AD-A034 239 SCHOOL OF AEROSPACE MEDICINE BROOKS AFB TEX F/G 9/2

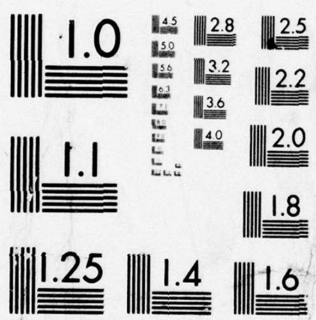
RETINAL THERMAL MODEL OF LASER-INDUCED EYE DAMAGE: COMPUTER OPE--ETC(U)

NOV 76 A R MERTZ, B R ANDERSON, E L BELL

UNCLASSIFIED SAM-TR-76-33 NL

1 OF 2

AD
A034239



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER SAM-TR-76-33	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) RETINAL THERMAL MODEL OF LASER-INDUCED EYE DAMAGE: COMPUTER PROGRAM OPERATOR'S MANUAL		5. TYPE OF REPORT & PERIOD COVERED Final Report June 1975 - June 1976
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Alan R. Mertz, Capt, USAF; Bruce R. Anderson, 1st Lt, USAF; Earl L. Bell, M.S.; David E. Egbert, Capt, USAF		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS USAF School of Aerospace Medicine (RAL) Aerospace Medical Division (AFSC) Brooks Air Force Base, Texas 78235		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62202F 6301-00-51
11. CONTROLLING OFFICE NAME AND ADDRESS USAF School of Aerospace Medicine (RAL) Aerospace Medical Division (AFSC) Brooks Air Force Base, Texas 78235		12. REPORT DATE November 1976
		13. NUMBER OF PAGES 136
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Ocular damage
Retinal thermal model
Laser effects
Temperature rise predictions

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A user-oriented description is given of a computer program for predicting thermal eye damage caused by exposure to laser radiation. This report describes the parameters necessary to run the program and provides a suggested range of values for the parameters. The input and output are described in detail, as are many of the capabilities and limitations of the program. The parameters and required input steps for a plot routine used to plot the temperature rises within the ocular structures are also presented. The architecture of the program and the interpretation of the input and output can be found in the IIT Research Institute technical report, "Thermal Model of Laser-Induced Eye Damage" (AD-A017-201).

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(6) (9) (10) (11) (12) 139p. (14) (16) (17) 00 317 000

TABLE OF CONTENTS

List of Illustrations

List of Tables

RETINAL THERMAL MODEL OF LASER-INDUCED EYE DAMAGE: COMPUTER PROGRAM OPERATOR'S MANUAL

INTRODUCTION

The Retinal Thermal Model is a mathematical model that predicts the thermal eye damage resulting from an exposure to laser radiation. This program, developed by the Illinois Institute of Technology Research Institute, is a result of many years of improvements in thermal damage modeling techniques. The mathematical basis for temperature predictions computed in the model is the standard heat-conduction equation in cylindrical coordinates

$$\left[\rho C\right]\frac{dv}{dt} = q(z,r,t) + K\left[\frac{1}{r}\frac{\partial v}{\partial r} + \frac{\partial^2 v}{\partial r^2}\right] + \frac{\partial}{\partial z}\left[K\frac{\partial v}{\partial z}\right]$$

where C = specific heat

ρ = density

q = rate of heat deposition from the laser

K = thermal conductivity

r = radial distance

z = axial distance

t = time

v = temperature rise above the initial temperature

The heat-conduction equation is approximated by finite differences and then solved with an explicit-implicit alternating-direction technique developed by D. W. Peaceman and H. H. Rachford (1). This technique solves the finite-difference equations explicitly in z and implicitly in r for odd time steps, and implicitly in z and explicitly in r for even time steps. In explicit calculations, existing temperatures are used to represent thermal gradients; in implicit calculations, future temperatures are used. This approach results in a set of equations that are solved using ordinary matrix algebra. Larger time intervals can be used with this technique than with standard explicit finite-difference methods. The model uses the predicted temperature rises to determine irreversible tissue damage by applying Henrique's damage criteria

$$\Delta\Omega(z,r,t) = C_1 \exp[C_2/T_a(z,r,t)]\Delta t$$

1. Peaceman, D. W., and H. H. Rachford, Jr. The numerical solution of parabolic and elliptic differential equations. J Soc Indust Appl Math 3:28-41 (1955).

where $\Delta\Omega(z,r,t)$ = incremental damage at point z, r

C_1 and C_2 = rate constants

$T_a(z,r,t)$ = absolute temperature

Δt = increment of time.

Irreversible tissue damage is defined as occurring whenever the integral of $\Delta\Omega$ over all time is greater than or equal to 1. From this mathematical basis, the model has the capability of predicting temperature rises, damage thresholds, and the extent of damage for specified sets of spatial coordinates within the ocular media. The model also has the capability to predict the retinal intensity distribution from the intensity distribution at the cornea. This optical spread capability has its basis in scalar diffraction theory, using the Fresnel approximation and adding terms to account for defocusing and ocular aberrations.

The Retinal Thermal Model has been divided into two programs, RE1 and RE2. Both programs perform the same tasks with one exception--RE1 contains the subroutine MXGRAN, which models the melanin granules, while RE2 does not contain MXGRAN.

Designed for maximum flexibility, the model offers wide variability in both input and output. It accommodates variations in laser radiation characteristics and in optical, thermal, and physiological properties of the eye. The model's design enables the user to specify his region of interest within the retinal layers and to print out only those portions of the output information which he desires.

The purpose of this manual is to give the user a basic understanding of the model's capabilities and how to use it within the limits of those capabilities. A meaningful description of a model of this type and flexibility, however, cannot be written without some complexity; and an individual will usually need some study and practical experience before feeling comfortable with the model. Additional information on the code can be obtained from the IITRI Technical Report, "Thermal Model of Laser-Induced Eye Damage" (2).

This manual briefly describes (1) the capabilities and limitations of the model as they pertain to the source, the eye, the mechanics of the program, and the output desired; (2) the basic input required, listing the required cards, their order, and appropriate formats; and (3) the printed output, including its format and the options available to the user. Appendix A is a glossary of all parameters that are either input or output, plus some parameters used internally in the program.

---

2. Takata, A. N., et al. Thermal model of laser-induced eye damage. Final Technical Report, IIT Research Institute Contract F41609-74-C-0005, 8 Oct 1974, USAF School of Aerospace Medicine, Brooks AFB, Tex. AD A017201.

Appendix B briefly describes the PLOT routine that can be used with the retinal model to obtain two- and three-dimensional plots of the predicted temperature rises. Appendix C covers the steps necessary to run the program on the IBM 360/65 computer at the San Antonio Data Service Center (SADSC), a computer facility available through a remote job-entry terminal located in the Biometrics Division, USAF School of Aerospace Medicine (USAFSAM), Brooks AFB, Tex. A description is included of the job-control language cards required to enter the program on the computer. Appendix D is a listing of the RE1 and RE2 programs and PLOT.

This manual is designed as a user's reference for the IITRI retinal model as it existed in November 1975. This version differs mainly in output format from the version described in the IITRI Technical Report.

CAPABILITIES AND LIMITATIONS

The user is responsible for adequately describing the exposure conditions to be modeled and the predictions (retinal intensity distribution, temperature rises, damage thresholds or extent of damage) he desires from the model. He must describe, or model, the incident radiation, the ocular media, the mechanics (temporal and spatial grid) of the program, and the output desired. This section presents a broad overview of the capabilities and limitations as they pertain to these four areas.

In developing the program, several major assumptions are made. First, the eye geometries are simulated in cylindrical coordinates, approximating the retina as a flat surface. Second, the relative retinal-intensity radial distribution is used at all depths of the eye below the retina, assuming that the incident radiation is coherent and dispersion of the beam through the retina will be minimal. Third, all reflected radiation is considered to move along axial directions; also, only first-order reflections are considered to be important to the total temperature rise. Fourth, the rates of retinal-tissue damage used in the damage integral are assumed to equate to the rates of skin-tissue damage; extensive work has been done with skin tissue in this area while very little has been done with the retina. Other assumptions will be discussed in later sections.

The model has a number of features which give the user flexibility in describing the incident radiation in terms of its spatial, spectral, and temporal properties. The model is designed only for monochromatic, coherent radiation. The spatial profile of the beam may be designated as uniform, gaussian, or irregular. Symmetry about the axis of propagation is always assumed. The user may specify the profile at either the cornea or the retina. For uniform or gaussian profiles, the user specifies the beam radius and the total power incident during a single pulse. For irregular profiles, the user constructs the desired beam profile by specifying the total power incident during a single pulse and the intensity (absolute or relative) and associated radial distances from the center of the beam.

The temporal properties of the incident radiation are specified by selecting the duration of a pulse, the repetition rate, and the number of pulses. Therefore, both single- and multiple-pulse exposures can be modeled. The model assumes all pulses to be square with respect to time; multiple-pulse trains will be composed of simple periodic, 100% modulated pulses. Because the model only predicts damage based on thermal mechanisms, the recommended time range for a pulse duration is between 10^{-7} and 10^{3} seconds. For pulse durations outside this range, nonthermal damage mechanisms may become significant. The model will accommodate shorter pulse durations; however, only thermal effects will be predicted. The model automatically converts pulses with durations less than 3×10^{-9} seconds to pulses with 3×10^{-9} -sec pulse widths. The conversion is accomplished assuming energy conservation. The associated power conversions are also made internally in the model. Even though the model makes these conversions, the output is always given relative to the original input with one exception: the value for POW, the total power incident on the corneal surface, is given relative to the 3×10^{-9} -sec pulse.

The ocular media can be described in terms of the optical, thermal, and physiological characteristics of the eye. The optical parameters include the coefficients for absorption, reflection, aberration, and refraction. Scattering of the radiation within the ocular media is ignored, and all reflected radiation is along axial directions. The user has the option to either specify an optical system that calculates the retinal intensity distribution or specify the retinal intensity distribution directly. The optical system of the eye is specified in terms of focal length, distances between optical surfaces, refractive indexes, aberration coefficients, and pupil size. The relative retinal-intensity radial distribution is assumed to remain constant at all depths below the retina.

The thermal parameters include heat capacity and conductivity of the individual ocular layers, initial ocular temperature, and blood flow within the choriocapillaris and tissues surrounding the eye. The user may also specify the decay of temperature rises within the melanin granules for program RE1.

The user specifies the physiological characteristics of the eye by selecting the radius and thicknesses of the various homogeneous ocular layers. All surfaces are assumed to be flat and perpendicular to the laser beam axis. The user may divide the pigment epithelium into two sublayers; also, the user has the option of placing 1-μm-cube melanin granules, separated by 1 μm, in either one of the sublayers of the pigment epithelium. The melanin granules are modeled as absorbing all energy incident upon them; the surrounding media absorb like the choroid. The choice of the sublayer to contain the melanin granules is used to differentiate between the human and the monkey eye. In the monkey eye, the melanin granules are located in the anterior portion of the pigment epithelium; in the human eye, in the posterior portion. The melanin granules contribute to the temperature rise calculations only for pulse widths less than 10^{-5} seconds. For times greater than 10^{-5} seconds, the heat diffusion from one granule to another has already taken place; so the temperature rises are

constant across the entire granulated layer, and the granules do not contribute significantly to the temperature calculations. For this reason, the Retinal Thermal Model has been divided into two programs, RE1 and RE2. RE1 contains the subroutine MXGRAN which models the melanin granules within the pigment epithelium. RE2 does not contain MXGRAN but does allow division of the pigment epithelium into two homogeneous layers. RE2 requires less computer central processing time and core memory than does RE1.

The model automatically selects the temporal and spatial grid points in the ocular media; however, the user specifies the size of the increments and extent of the temporal and spatial grid. The dimensions of the spatial and temporal increments represent the limits of resolution of the model predictions. The spatial grid has uniform increments in the region of highest temperature rise and constantly expanding increments away from the highest temperature regions. The temporal grid has constantly expanding time steps from the beginning of the pulse. The predicted retinal intensity distribution and temperature rises are all relative to the specified spatial and temporal grid points. The user selects the range and spacing of spatial coordinates used to print the temperature rises, threshold powers, and extent of damage; also, the user has the option to print and plot temperature rises at any selected time. A separate plotting routine uses input data cards punched by the retinal model and user control cards to plot the temperature rises. Entire sections of the complete printed output can be deleted. Available options are described in the Output Format section.

The retinal model has a variety of input/output capabilities. The user can batch a sequence of exposures by varying the pulse-repetition frequency and/or number of pulses while keeping all other parameters constant. For such sequence of exposures, the program is initiated only once, thus conserving operating time.

INPUT REQUIREMENTS

This section describes the input required to specify the source, the ocular media, the mechanics of the program, and the formatting of the output. All parameters required by the user as input, along with the appropriate formats and data card numbers, are given for retinal models RE1 and RE2. All parameters are defined in Appendix A. When solving a problem, the user must model both the incident radiation and the ocular media to fit the desired situation, and must also specify the parameters governing the mechanical operation of the program, such as the size of the grid required and the output desired.

To simulate the radiation incident on the ocular media, the user has the option of specifying, via IPROF, a uniform-, gaussian-, or irregular-beam irradiance profile. Axial symmetry for all beam profiles is assumed. For uniform-beam profiles, the beam radius (RIM) must be specified; it is used with LIM (the number of radial intervals from the center of the beam to a specified radius) to establish the minimum radial grid increment (DR). For gaussian-beam profiles, RIM must be specified at a particular relative

intensity point (CUT). For irregular-beam profiles, the absolute or relative irradiance profile must be specified on a point-by-point basis by listing the irradiance value, PX(L), at the radial distance, RX(L), from the center of the beam. PX(L) cannot have a value of zero at the center of the beam. The total number of specified irradiance points is equal to LR. The model will integrate the beam profile at radial intervals (RINT). We suggest that the irradiance points be specified at radial intervals equal to multiples of RINT to avoid interpolation.

The total power per pulse in the beam at the cornea (POW) must be specified for all beam profiles. In effect, the user has the capability of specifying the divergence, which is a function of the distance of the pupil from the nearest beam waist (ZO). ZO is input only if the spread function is used (IFIL=1). In addition, the pulse width (DPULSE), the pulse repetition frequency (REPET), the number of pulses entering the ocular media (NPULSE), and the wavelength (WAVEL) must also be specified for all simulated exposures. However, WAVEL is used only in the image spread-function calculation (IFIL=1). For single-pulse exposures, a value for REPET must be supplied; however, it will not be used in calculations in the model.

The eye is modeled as a cylinder with its axis coincident with the axis of radiation propagation. The various layers of the eye lie perpendicular to the cylinder axis, with flat boundaries between the layers. The radial extent of the eye is specified as RVL. The various ocular media modeled in the eye are listed in Table 1. The thickness, transmittance, reflectance, and absorption coefficients of the various layers are all input. The user must also supply the thermal conductivity, CONX(L), and heat capacity, VSHX(L), for each ocular layer. Values for these parameters are given in Appendix A.

TABLE 1. OCULAR STRUCTURES MODELED

Label	Ocular structures
1	Everything from the anterior portion of the eye to the pigment epithelium
2	Pigment epithelium
3	Choriocapillaris (vascular layer)
4	Choroid
5	Sclera
6	Tissue posterior to the sclera

The pigment epithelium layer may be divided into two sublayers, with the user specifying the thickness and the absorption coefficient for each sublayer. IGX is the parameter used to specify the absorption coefficients. For a simulation of the human eye (IGX=1), the absorption coefficient for the anterior sublayer (APE1) is set equal to the absorption coefficient for the choroid (ACH); the absorption coefficient for the posterior sublayer (APE2) is calculated within the model. For the simulation of the monkey eye (IGX=0), the absorption coefficient for APE1 is calculated within the model, while APE2 is set equal to ACH. The distinction of two sublayers with differing absorption coefficients is due to the assumption that most of the absorption within the pigment epithelium occurs in the posterior section for human eyes and in the anterior section for monkey eyes. The relative thickness of the two sublayers is specified by RPE.

The user can transform the distribution of the beam incident on the cornea to a retinal distribution by using the spread function (IFIL=1) or can ignore that transformation (using IFIL=0) and specify the intensity distribution at the retina. The spread function simply transfers the beam from the cornea to the retina, simulating ocular focusing and optical aberration effects. The initial temperature of the eye is specified as T0. The radius of the pupil of the eye (PUPIL) is specified by the user and implemented within the model to define the initial beam intensity profile. Only when the spread function is used must Z0, FL0, FC, NB, PP, CABER, PC, J0, and NA be input.

The effects of blood flow are assessed in two ocular structures--the choriocapillaris and the tissue surrounding the eye. Within both structures, the blood is treated as a heat sink--the extraction of heat from the adjacent tissue by the blood as it enters that tissue. The user inputs the specific heat of blood (SHB), the total blood flow to the choriocapillaris (CFLOW), and the rate of blood flow to the tissue surrounding the eye (XFLOW). The model computes the temperature rise resulting from the heating of the blood as it enters both ocular structures. The user can also account for the radial transport of heat by the radial flow of blood within the choriocapillaris. To do this, statement number 31 of the RE1 and RE2 programs (Appendix D) must be deleted and replaced by statements to establish specific values for XFLOWO(L1) L1=1,6. XFLOWO(L1) is defined as the total blood flow per unit area leaving the choriocapillaris at a given radial distance R. It is given in units of $g \cdot cm^{-2} \cdot sec^{-1}$. Without this change, the model will only treat blood as a heat sink.

The model also computes the temperature effects of heat absorption in the melanin granules. The user must specify TS(L), which determines how the average temperature rise of the granules decays with time, and LTMAX, which controls the time beyond which the temperature rises of the melanin granules are completely dissipated. The temperature-rise contributions are specified at time increments of 3×10^{-9} seconds. In the output section of the model, XPD(K) represents the degree to which the melanin granules affect the temperature in the pigment epithelium.

To predict the power required to produce irreversible damage by the damage-integral method, the coefficients for the rates of damage (DAMAGE) must be specified. The model will also compute the power required to raise the temperature of the tissue to TSTEAM and will repeat the calculation at temperature intervals (DTSTM) until the power to produce irreversible damage, as determined by the damage-integral method, is reached.

The mechanics of model operation include determining the spatial grid system and the time intervals used in the temperature-rise computation. The spatial grid system, used to specify the locations at which an evaluation of the temperature-rise and damage-threshold predictions is desired, has both a uniform and a constantly expanding portion. The uniform portion of the grid is positioned at the center of the beam in the pigment epithelium, usually the region of highest temperature rise. The grid then constantly expands away from this region. The user specifies the size of the uniform radial grid interval (DR) by using LIM and LESION for irregular or gaussian profiles, and LIM and RIM for uniform profiles. The uniform axial grid interval is about one-sixth of the thickness of the pigment epithelium. Upon this grid the physiological layers are constructed. The various ocular layers and the labels used to assign absorption, conductivity, and heat-capacity values to these ocular layers are listed in Table 1. For damage threshold calculations, the range of grid locations at which calculations are made is determined from LIMAX and MAXPRT for axial locations and from RMAX for radial locations. LIMAX and RMAX must be chosen so that (ID2-ID1+1)* JM $\leq$ 27 where ID1=IMAX-LIMAX, ID2=IMAX+LIMAX, and JM is the index of the first radial grid point beyond RMAX. IMAX is the axial grid point at which peak temperature rises occur at the conclusion of the pulse.

The time intervals used in calculating temperature rises, the maximum time during which temperature rise calculations and damage threshold predictions are made, and the time intervals used to subdivide the pulse widths are complex and intricately related. Unless the reader is experienced with the program, we suggest that the values supplied in Appendix A for FTIME, EDT1, EDT2, NPT, XCT, and KTT be used. Appendix A contains some of the relationships between these parameters for those who need to change the suggested values.

To reduce computation time of the program, the user may group all exposures which differ only in REPET and NPULSE. This is done by specifying the desired values for REPET and NPULSE and specifying the total number of exposures so grouped in NTEST, with a maximum of 7 pairs of values per group.

The formatting of the output includes selecting the output sections to be printed and the times and spatial ranges within the grid system for temperature rise printouts and/or plotting. Using IPRT codes, as shown in Table 2, the user specifies the output sections desired. For multiple-pulse exposures, temperature rises are printed only for the first pulse

TABLE 2. PROGRAM OUTPUT SECTIONS

Code	Section
IPRT(1)	Grid information
IPRT(2)	Laser profile
IPRT(3)	Data identification
IPRT(4)	Blood flow and heat deposition
IPRT(5)	Temperature rises
IPRT(6)	Normalized temperature rises
IPRT(7)	Normalized temperature rises of melanin granules
IPRT(8)	Predicted threshold laser power
IPRT(9)	Axial extent of damage
IPRT(10)	Radial extent of damage

incident on the retina; however, prediction of damage is based on multiple-pulse effects. The range of grid locations at which temperature rise calculations are printed is determined from ID1 and ID2 for the axial range, and from JD1 and JD2 for the radial range. The user has the option of printing the temperature rises at all the time intervals determined within the model (ITYPE=1), only every nth time interval (ITYPE=n), and/or at any selected times (KTYPEO=1) within the maximum time used by the model. The total number of selected times for printing is equal to KTYPE, while the selected times for printing are specified in TIMEX. Temperature rise calculations are always printed at the beginning and end of the pulse and at the time interval TIME. When plots of temperature rises at selected times are desired (KTYPEO=0), the model will always provide printouts in addition to the plots. The range of axial and radial grid locations desired for plots is determined by II1, II2, and JJ1, JJ2, respectively. II3 enables the user to mark (by an asterisk) a specific axial depth on plots for easy identification and comparison with other plots.

Tables 3 and 4 provide a quick reference to the parameters required as input to the model as well as their required formats and order. Data cards with an asterisk as a prefix to the data card number should be checked to ensure they simulate the desired exposure; data cards without the prefix asterisk generally remain constant from exposure to exposure. Data cards with an asterisk as a suffix to the data card number are not input unless an irregular profile (IPROF=2) or the spread function (IFIL=1) is used. When the uniform and gaussian profiles are used (IPROF=0,1), data cards 20, 21, and 22 are not input. When the spread function is not used (IFIL=0), data cards 23, 24, and 25 are not input.

TABLE 3. INPUT DECK FOR RE1

Data card number	Format	Parameter
1 (4 cards)	11F7.2	FTIME(L) L=1,38
*2 (1 card)	10I7	IPRT(I) I=1,10
*3 (1 card)	F7.4,3I7	RIM, LIM, IFIL, IGX
*4 (1 card)	F7.2,2I7,F7.2	RMAX, LIMAX, MAXPRT, LESION
*5 (1 card)	I7,3E7.2	IPROF, POW, CUT
*6 (1 card)	10E7.2	DPULSE
*7 (1 card)	10I7	NTEST, NRUN(L) L=1, NTEST
*8 (1 card)	10E7.2	REPET(L) L=1, NTEST
*9 (1 card)	10I7	NPULSE(L) L=1, NTEST
*10 (1 card)	10I7	ID1, ID2, JD1, JD2, ITYPE
11 (1 card)	11F7.2	TO, EDT1, EDT2
*12 (1 card)	11F7.2	TOM, APE, AVL, ACH, ASC, ATS, RCO, RRT, RSC, RPE, WAVEL
*13 (1 card)	11F7.2	TAV, TPE, TVL, TCH, TSC, RVL
14 (1 card)	11F7.2	CONX(L) L=1,6
15 (1 card)	11F7.2	VSHX(L) L=1,6
16 (4 cards)	10I7	NPT(L) L=1,38
17 (4 cards)	10F7.3	XCT(L) L=1,38
18 (4 cards)	10I7	KTT(L) L=1,38
19 (1 card)	10E8.3	PUPIL
20(1 card)	I7	LR
21(1-3 cards)	10E7.3	RX(L) L=1,LR
22(1-3 cards)	10E7.3	PX(L) L=1,LR
23(1 card)	10E8.3	ZO, FLO, FC, NB, CABER, PP, PC
24*(4 cards)	10F8.5	JO(L) L=1,32
25*(3 cards)	10F8.5	NA(L) L=1,22
26 (1 card)	10F7.3	SHB, XFLOW, CFLOW
*27 (1 card)	I7,3E7.2	KTYPEO
*28 (1 card)	I7,3E7.2	KTYPE
*29 (1 card)	10E7.2	TIMEX(K) K=1,KTYPE
*30 (1 card)	10I7	II1, II2, II3, JJ1, JJ2
31 (1 card)	I7,3E7.2	LTMAX
32 (22 cards)	10F7.3	TS(L) L=1,LTMAX, 10
33 (1 card)	11F7.2	DAMAGE(L2,1) } L2=1,2 DAMAGE(L2,2) } TSTEAM, DTSTM

Prefix * indicates parameters most often altered for specific exposures.

Suffix * indicates parameters not input unless irregular profile or spread function is used.

TABLE 4. INPUT DECK FOR RE2

1-30	Same as RE1	Same as RE1
31(1 card)	11F7.2	DAMAGE(L2,1) } L2=1,2 DAMAGE(L2,2)

OUTPUT FORMAT

The printed output of the retinal models, RE1 and RE2, are arranged into 11 sections. The printout format and the IPRT codes for each section are listed in Table 5. The user has the option of printing only the sections desired, as specified in Table 2, except for one. The unlabeled section listed prior to Predicted Threshold Laser Power on Table 5 is always printed. (The definitions for the program parameters are given in Appendix A.) Not every parameter in Table 5 will be printed every time. In the Laser Profile Section, if the spread function is not used, all parameters dealing with it will not be printed. On the first line of this section, only RIM is printed for a uniform beam (IPROF=0); SIGMA, RIM, and CUT for a gaussian beam (IPROF=1); or RINT for an irregular beam (IPROF=2). For a single-pulse exposure (N=1), trainlength and repetition rate are deleted from the unlabeled section.

In the Temperature Rises section (Table 5), numbers printed at each axial and radial grid point represent the temperature rise (°C) above the initial temperature of the eye (TO) at the time indicated. For the Normalized Temperature Rises section, the temperature rise (°C) is divided by the input power. Therefore, the numbers printed at each grid point represent the temperature rise per watt of input power. For pulses of less than a 3×10^{-9}-sec duration, POW is converted to a power relative to a 3×10^{-9}-sec pulse; the converted POW is used to normalize the temperature rises.

For the axial and radial extent of damage, the model selects the appropriate statement from those listed in Table 5. If the input power produces no damage within the grid range specified by LIMAX, the model will print NO DAMAGE--LASER POWER TOO LOW. If the grid range specified by LIMAX contains the most anterior point at which damage occurs, the model will print MINIMUM DEPTH OF DAMAGE = (the value given will be relative to the anterior boundary of the pigment epithelium). If the maximum posterior point at which damage occurs is contained within the grid range specified by LIMAX, the model will print MAXIMUM DEPTH OF DAMAGE = (again, the value given will be relative to the anterior boundary of the pigment epithelium). If damage is present, the radial extent of damage will be printed for each axial grid point specified by LIMAX. The value for the radial extent of damage will be relative to the center of the laser beam.

TABLE 5. PRINTED OUTPUT FOR RE1

GRID INFORMATION IPRT(1)

R2 =					
R1 =	ZM =				
ID1 =	ID2 =	JD1 =	JD2 =		
DR =	DZ =				
IPA =	IPC =	IPE =	IPS =	IPT =	IPV =
LPA =	LPC =	LPE =	LPS =	LPV =	
M =	M1 =	N =	N1 =		
R =					
Z =					
ZH =					

LASER PROFILE IPRT(2)

SIGMA =	RIM =	CUT =	RINT =
Z0 =		FLO =	
CABER =	CABER2 =	PP =	
PC =	NB =	NC =	
FC =	WAVEL =		
QP =			
HR =			

DATA IDENTIFICATION IPRT(3)

REPET =				
NPULSE =				
AAV =	ACH =	APE =	ASC =	ATS =
RCO =	RRT =	RPE =	TOM =	AVL =
TAV =	TCH =	TPE =	TSC =	TVL =
IGX =	IFIL =	IPROF =	LIM =	NTEST =
POW =	DPULSE =	RIM =	RMAX =	TIME =
CFLOW =	XFLOW =	SHB =	EDT1 =	EDT2 =
DT =	KM =	KT =	PTIME =	XC =
IKX =	AP =	APE1 =	APE2 =	IG =
RVL =	PUPIL =	TO =	LIMAX =	MAXPRT =

BLOOD FLOW AND HEAT DEPOSITION IPRT(4)

FLOWI =
FLOWX =
S =
S =
.
.
.
S =

TABLE 5. (Continued)

TEMPERATURE RISES   IPRT(5)

```
    TIME =      K =
                R =
    Z =
    Z =
    .
    .
    .
    Z =
```

NORMALIZED TEMPERATURE RISES   IPRT(6)

```
    TIME =        K =        POWER = 0.1000E+01 WATTS
                  R =
    Z =
    Z =
    .
    .
    .
    Z =
```

NORMALIZED TEMPERATURE RISES OF MELANIN GRANULES   IPRT(7)

```
    LTMAX =       BT =
    XPD =
```

(unlabeled section)

```
    WAVELENGTH =        DAMAGE =
    NRUN =              TRAINLENGTH =     SEC       PULSE WIDTH =       SEC
    NUMBER OF PULSES =          REPETITION RATE =               PULSES/SEC
    IMAGE RADIUS =              LESION RADIUS =        CM
```

PREDICTED THRESHOLD LASER POWER   IPRT(8)

```
    R =
    Z =        QD =
    Z =        QD =
    .
    .
    .
    Z =        QD =
```

TABLE 5. (Continued)

<u>TEMPERATURE RISES AT SELECTED TIMES</u> TIMEX(K)

TIME =
R =
Z =
Z =
.
.
.
Z =

<u>AXIAL EXTENT OF DAMAGE</u> IPRT(9)

NO DAMAGE--LASER POWER TOO LOW
or
DEPTHS OF DAMAGE BEYOND BOTH SPECIFIED DEPTHS
or
MINIMUM DEPTH OF DAMAGE = CM
and/or
MAXIMUM DEPTH OF DAMAGE = CM

<u>RADIAL EXTENT OF DAMAGE</u> IPRT(10)

Z = CM RADIAL EXTENT OF DAMAGE GREATER THAN CM
or
Z = CM RADIAL EXTENT OF IRREVERSIBLE DAMAGE = CM

APPENDIX A

GLOSSARY

All parameters used as either input or output in the retinal models, and some used internally, are listed in alphabetical order, with appropriate units and suggested input values. The equations provided are in FORTRAN IV language, where ALOG represents the natural logarithm and ** represents "raised to the power." For some of the parameters, numerical values are tabulated in the tables at the end of the glossary.

AAV--The absorption coefficient for the ocular media from the cornea to the retina.

Units: inverse cm

AAV = ALOG(TOM)/TAV

ACH--The absorption coefficient for the choroid.

Units: inverse cm
Suggested input value: Tables A-1 and A-2

AP--The fraction of heat that, deposited in the granulated pigment epithelium, is absorbed by the melanin granules. AP is calculated from ACH, RPE, TPE, APE1, and APE2. It is printed and used only when the subroutine MXGRAN in RE1 is used.

Units: unitless

APE--The absorption coefficient of the pigment epithelium.

Units: inverse cm
Suggested input value: Tables A-1 and A-2

APE1--The absorption coefficient for the anterior sublayer of the pigment epithelium.

Units: inverse cm

APE1 = (APE-ACH*(1.-RPE))/RPE for IGX=0
APE1 = ACH for IGX=1

APE2--The absorption coefficient of the posterior sublayer of the pigment epithelium.

Units: inverse cm

APE2 = ACH for IGX=0
APE2 = (APE-ACH*RPE)/(1.-RPE) for IGX=1

ASC--The absorption coefficient for the sclera.

Units: inverse cm
Suggested input value: same as ACH

ATS--The absorption coefficient for the tissue posterior to the sclera of the eye.

Units: inverse cm
Suggested input value: same as ACH

AVL--The absorption coefficient for the choriocapillaris (vascular layer).

Units: inverse cm
Suggested input value: same as ACH

BT--The time interval during which heat conduction from the granules is insignificant. It is the time interval used to evaluate the contributions of the melanin granules to the temperature rises. BT is set equal to 0.3×10^{-8}.

Units: sec

CABER--A constant in the spherical aberration term used in the spread function. The spherical aberration term is CABER ρ^4/λ, where ρ is the radius in the pupil plane and λ is the wavelength. CABER is printed only when the spread function is used (IFIL=1).

Units: $cm^{-4} \cdot nm$
Suggested input value: -3.0E+6

CABER2--A spherical aberration constant calculated by dividing CABER by the wavelength (nm) of the incident radiation. CABER2 is printed only when the spread function is used (IFIL=1).

Units: cm^{-4}

CFLOW--The total blood flow to the choriocapillaris.

Units: $g \cdot sec^{-1}$
Suggested input value: 0.024

CONX(L),L=1,6--The thermal conductivity of the Lth ocular media as listed in Table 1 (text).

Units: $cal \cdot cm^{-1} \cdot sec^{-1} \cdot {}^{\circ}C^{-1}$
Suggested input value: 0.0012

CUT--The fraction of the peak intensity in the beam cross-sectional distribution at which the beam radius, RIM, is specified for gaussian profiles (IPROF=1). CUT can be any fraction of the peak intensity, but RIM must be specified at the same point.

Units: unitless
Suggested input value: 1.35E-1 ($1/e^2$ intensity points of a gaussian profile)

DAMAGE (L2,1), DAMAGE (L2,2), (L2=1,2)--The DAMAGE array contains the coefficients for the damage-rate integral.
For temperatures below 50°C:
Rate = EXP (DAMAGE(1,1) - DAMAGE(1,2)/(VC+273+TO)).
For temperatures above 50°C:
Rate = EXP (DAMAGE(2,1) - DAMAGE(2,2)/(VC+273+TO)).
VC is the temperature rise (°C) at the specified grid points. TO is the initial temperature (°C) of the eye, and the number 273 converts degrees Celsius to Kelvin. The values provided are for skin tissue; but they are assumed to equate to the damage-rate constants for retinal tissue.

Units: DAMAGE(L2,1): unitless
DAMAGE(L2,2): sec^{-1}
Suggested input values: DAMAGE(1,1) = 149.
DAMAGE(1,2) = 50,000.
DAMAGE(2,1) = 242.
DAMAGE(2,2) = 80,000.

DPULSE--The exposure duration of an individual pulse.

Units: sec
Suggested input value: 3.0E-9 to 1.0E+3

DR--The radial increment in the uniform portion of the grid network.

Units: cm

DR = LESION/LIM for gaussian and irregular beam profiles (IPROF=1,2)
DR = RIM/(LIM-.5) for uniform profiles (IPROF=0)
NOTE: For IFIL=1, since RIM is a corneal dimension, LIM must be large to obtain a small DR.

DT--The initial time interval after the start of a pulse at which temperature rise calculations are made. Successive calculated time intervals are increased by the stretching factor XC.

Units: sec

DT = DPULSE * (XC-1.)/(XC**NP-1.)
NP = NPT(L1) for single pulse
XC = XCT(L1) for single pulse
L1 = ALOG(DPULSE)/.69315 + 29.
NP = 5 for multiple pulse
XC = 1.4 for multiple pulse

DTSTM--The temperature increment used to increase TSTEAM in calculating the power required to produce the temperature TSTEAM in the melanin granules. Successive calculations and printouts of Predicted Threshold Power will be made at each increment of TSTEAM until the power required to produce the temperature TSTEAM exceeds the power required to produce irreversible damage as predicted by the damage integral method. Reducing DTSTM results in increased computation time and printout.

Units: °C
Suggested input value: 200.

DZ--The axial increment in the uniform portion of the grid network.

Units: cm

DZ = TPE/M1 - 1.E-25
M1 = 6

EDT1, EDT2--Parameters used to determine the time intervals at which temperature rises are calculated. The model divides the computed time interval into 2*IKX subintervals to insure stability and accuracy. IKX is dependent upon TIME, EDT1, and EDT2. The suggested values are adequate except for pulse widths greater than 10^3 sec.

Units: unitless
Suggested input values: EDT1 = 0.16; EDT2 = 1.

IKX = TIME**EDT1 + EDT2

FC--The focal length of the cornea measured in the ocular media. FC is required only when the spread function is used (IFIL=1).

Units: cm
Suggested input value: 3.12EØ--humans
2.43EØ--rhesus monkeys

FLO--The second principal focal length at a 500-nm wavelength. The second principal focal point is the point at which parallel light incident upon the anterior portion of the eye will focus. FLO is required only when the spread function is used (IFIL=1).

Units: cm
Suggested input value: 2.242EØ--humans
1.684EØ--rhesus monkeys

FLOWI(J), J=1,JVL--The flow of blood into a unit volume of the choriocapillaris at some radial point, R(J). JVL is the index such that R(JVL)=RVL.

Units: $g \cdot cm^{-3} \cdot sec^{-1}$

FLOWX(J), J=1,JVL--The product of the radius at some radial point, R(J), and the net flow of blood per unit area in the radial direction at point R(J). JVL is the index such that R(JVL)=RVL.

Units: $g \cdot cm^{-1} \cdot sec^{-1}$

FTIME(L), L=1,38--The array FTIME is used for multiple-pulse exposures to determine the time interval (TIME) over which the damage integral is evaluated. TIME is a function of FTIME; each element of FTIME is associated with a range of pulse widths (DPULSE).

Units: unitless
Suggested input value: 1.8 for all elements

TIME = FTIME(L1) * X1 for multiple pulse
L1 = ALOG(DPULSE)/.69315 + 29.
X1 = NPULSE(L)/REPET(L) The largest value for any NTEST.

Therefore, to increase the time interval (TIME) over which the damage integral is evaluated, one should increase the Lth element of FTIME associated with the specified pulse width (DPULSE).

HR(J), J=1,N--The normalized retinal irradiance at radial position R(J). Symmetry about the axis is assumed.

Units: unitless

ID1, ID2--Input and output parameters. As input parameters, ID1 and ID2 are integers used to determine at what axial positions the temperature rises are to be printed. The temperature rises will be printed at axial positions with indexes from I=IPE+ID1 to I=IPE+ID2. As output parameters, ID1 and ID2 are actual grid index points relative to the first grid point located anterior to the cornea. Temperature rises are to be printed from grid point ID1 to point ID2.

Units: unitless
Suggested input values: dependent upon the user

ID1 (output) = IPE + ID1 (input)
ID2 (output) = IPE + ID2 (input)

IFIL--The parameter that allows the user to decide whether or not to use the spread function. The spread function is used to transfer the beam distribution from the cornea through the ocular media to the retina. When the spread function is used, all input (RIM, POW, CUT, RX(L), PX(L), LR, ZO) beam characteristics must apply to the beam at the cornea. When the spread function is not used, the input spatial beam characteristics are assumed to apply to the beam distribution at the retina, with the exception of POW and PX(L) which always apply to the cornea.

Units: unitless
Suggested input value: 1--spread function is used.
0--spread function is not used.

IG--The index of the initial grid point in the melanin granules. It is used and printed only in program RE1.

Units: unitless

IGX--The selection parameter for the absorption coefficients of the two sublayers modeled in the pigment epithelium. For IGX=1, the absorption coefficient for the anterior sublayer (APE1) is equal to ACH. The absorption coefficient for the posterior sublayer (APE2) is computed assuming it contains most of the melanin granules. For IGX=0, APE2=ACH and APE1 is computed assuming the anterior sublayer contains most of the melanin granules.

Units: unitless
Suggested input value: 1--a human eye
0--a monkey eye

II1, II2--The indexes used to specify the range of axial grid values desired for a plot. These indexes are the actual indexes of grid points, with II1 closer to the anterior part of the eye.

Units: unitless
Suggested input values: dependent upon the user

II3--An identification index used in the plotting routine. An asterisk can be placed on the curve at the axial depth associated with grid point II3 in a plot and allows easy reference for comparing similar curves in more than one plot. II3 is the index of an actual grid point.

Units: unitless
Suggested input value: dependent upon the user

IKX--The number of times the temperature calculations are repeated to insure stability.

Units: unitless

```
IKX = TIME **EDT1 + EDT2
TIME = FTIME(L1)*X1 for multiple pulse
L1 = ALOG (DPULSE)/.69315 + 29.
X1 = NPULSE/REPET the largest value for any NTEST
TIME = DT * (XC **KT -1.)/(XC-1.) for single pulses
KT = KTT(L1)
XC = XCT(L1)
```

IPA--The index of the initial grid point located in the cornea. Its value is always 2.

Units: unitless

IPC--The index of the initial grid point in the choroid.

Units: unitless

IPE--The index of the initial grid point in the pigment epithelium. Its current value is 10.

Units: unitless

IPROF--The parameter used to describe the laser intensity profile. If a uniform profile is specified, RIM and POW must be specified. For a gaussian profile, RIM, CUT, and POW must be given. Irregular profiles require PX(L), RX(L), LR, and POW.

Units: unitless
Suggested input value: 0--uniform profile
1--gaussian profile
2--irregular profile

IPRT(I),I=1,10--The parameter which gives the user the choice of printing or not printing each of 10 separate output sections described in text Output Format section.

Units: unitless
Suggested input value: 0--printing is not desired.
1--printing is desired.

IPS--The index of the initial grid point in the sclera.

Units: unitless

IPT--The index of the initial grid point in the tissue posterior to the sclera.

Units: unitless

IPV--The index of the initial grid point in the choriocapillaris.

Units: unitless

ITYPE--Used to determine the time indexes (K) at which the temperature rises will be printed. The total number of times the temperature rise calculations can be printed is equal to KT. If the temperature rises are to be printed at all times (XT(K) K=2,KT), ITYPE must equal one. If temperature rises are to be printed at every nth time, ITYPE must equal n. Temperature rise printouts will always be provided at the first time (K=2) after initiation of the pulse, at the conclusion of the pulse (K=KM), and at the final time over which damage is assessed (TIME=XT(KT)). ITYPE must never equal zero.

Units: unitless
Suggested input value: dependent upon the user

JD1, JD2--The radial indexes used to determine the radial positions from the center of the laser beam at which the temperature rises are to be printed. The model will print the temperature rises starting at radial position JD1, out to radial position JD2. JD1=1 corresponds to the z-axis or the center of the beam. All 14 radial grid points can be printed; but only 9 will be printed on a single line, with the other 5 printed in consecutive order on the second line.

Units: unitless
Suggested input values: dependent upon the user

JJ1, JJ2--The indexes of grid points used to specify the range of radial grid values desired for a plot. JJ1 is the index closer to the center of the beam.

Units: unitless
Suggested input values: JJ1 = 1, and JJ2 = 5

JO(L), L=1,32--The value of the zero-order Bessel function for argument values to 3.1 in 0.1 increments. It is used in constructing the spread function.

Units: unitless
Suggested input values: Table A-6

K--An index of the expanded times, XT(K)--times at which temperature rise calculations are made.

Units: unitless

KM--The index indicating the temperature rise printout occurring at the end of the pulse (XT(KM)=DPULSE).

Units: unitless

KT--The maximum number of times at which temperature rise calculations are computed.

Units: unitless

XT(KT) = TIME
KT = KTT(L1) for single pulse
L1 = ALOG(DPULSE)/.69315 + 29.
KT = [ALOG(1.+TIME*(XC-1.)/DT)/ALOG(XC)+1.]+1 for multiple pulses

KTT(L) L=1,38--An array of the number of steps used to reach the total time (TIME). The suggested values were calculated to reduce error and increase stability in solving the finite-difference equations in the model.

Units: unitless
Suggested input values: Table A-4

KTYPE--The total number of temperature rise plots or selected time printouts desired. If no plots or selected time printouts are desired, set KTYPE=0. KTYPE has a maximum value of 10. A printout of the temperature rise is automatic with each requested plot.

Units: unitless
Suggested input value: dependent upon the user

KTYPEO--The parameter that controls the punching of data cards used as input to the plot routine.

Units: unitless
Suggested input value: 0--card punching
1--no card punching

LESION--The radius of the retinal lesion. It is used only to determine DR for efficient grid structuring. It is not used for uniform beam profiles (IPROF=0).

Units: cm
Suggested input value: dependent upon the user

DR = LESION/LIM for gaussian and irregular beam profiles (IPROF=1 or 2).

LIM--The number of radial intervals from the center of the beam to RIM for uniform beam profiles (IPROF=0), or to LESION for gaussian and irregular profiles (IPROF=1 or 2). LIM is used to determine the size of the smallest uniform radial-grid increment (DR). There are only four uniform radial-grid intervals.

Units: unitless
Suggested input value: 5

LIMAX--A parameter that determines the range of axial distances at which damage calculations are printed. It is used in conjunction with MAXPRT. For single-pulse exposures, LIMAX has a maximum value of 9 for MAXPRT=2 or 3, and a maximum value of 4 for MAXPRT=1. For multiple-pulse exposures, LIMAX has a maximum value of 2 regardless of MAXPRT.

For MAXPRT = 1, axial distance = IMAX - 2 LIMAX to IMAX
MAXPRT = 2, axial distance = IMAX $\pm$ LIMAX
MAXPRT = 3, axial distance = IMAX to IMAX + 2 LIMAX
IMAX = the axial grid point at which peak temperature rises occur at the conclusion of the pulse.

Units: unitless
Suggested input value: dependent upon the user

LPA--The index of the last grid point located in the vitreous humor.

Units: unitless

LPC--The index of the last grid point in the choroid.

Units: unitless

LPE--The index of the last grid point in the pigment epithelium.

Units: unitless

LPS--The index of the last grid point in the sclera.

Units: unitless

LPV--The index of the last grid point in the choriocapillaris.

Units: unitless

LR--The total number of profile values to be specified (LR has a maximum value of 30). For irregular beam (IPROF=2) distributions only, the user must specify the intensity distribution of the beam on a point-by-point basis by giving the profile irradiance value, PX(L), and associated radial distances, RX(L).

Units: unitless
Suggested input value: dependent upon the user

LTMAX--The parameter that controls the time beyond which the temperature rises of the melanin granules are completely dissipated. LTMAX must be large enough to allow the temperature rises in the melanin granules to decrease to an insignificant value. The suggested value has been found to be adequate, and it is recommended that LTMAX not be less than 2191.

Units: unitless
Suggested input value: 2191

M--The total number of grid spaces in the axial direction, an even integer; currently, M = 28.

Units: unitless

M1--Half the number of uniformly spaced axial increments; currently M1 = 6.

Units: unitless

MAXPRT--The parameter which gives the user the option to control the printing of the predicted threshold laser powers and extent of damage. If MAXPRT equals 1, predicted threshold laser power calculations will be printed only at axial positions anterior to the position of the peak temperature rise, IMAX(IMAX-2 LIMAX to IMAX). If MAXPRT equals 2, the printouts will be for axial positions both anterior and posterior to the peak temperature rise position (IMAX-LIMAX to IMAX + LIMAX). If MAXPRT equals 3, printouts will be made only for axial positions posterior to IMAX (IMAX to IMAX + 2* LIMAX).

Units: unitless
Suggested input value: 1--anterior side of peak temperature
2--both sides of peak temperature
3--posterior side of peak temperature

N--The total number of grid spaces in the radial direction; currently, N=13.

Units: unitless

N1--The number of uniform grid increments in the radial direction; currently N1=4.

Units: unitless

NA(L), L=1,22--The refractive index of the ocular media as a function of wavelength. They should be placed on the data card in increasing wavelength sequence from 350 nm, at 50-nm increments.

Units: unitless
Suggested input values: Table A-7 (for water)

NB--The index of refraction of the ocular media at a 500-nm wavelength. NB is required only if the spread function is used (IFIL=1).

Units: unitless
Suggested input value: 1.336EØ(mainly for water)

NC--The index of refraction of the ocular media for wavelength (WAVEL). NC is printed only if the spread function is used (IFIL=1).

Units: unitless

NP--Constant used within the program.

Units: unitless

NPT(L), L=1,38--The number of incremental times used to subdivide DPULSE. It is associated with specific values of FTIME(L), XCT(L), and KTT(L), all of which are associated with a specific range of values of DPULSE and DT. The suggested values were calculated to keep the errors small and satisfy stability requirements for solving the heat-conduction boundary value problem through the use of finite differences.

Units: unitless
Suggested input values: Table A-4

NPULSE(L), L=1, NTEST--The number of pulses associated with a specified test exposure identified by NRUN(L). All other parameters except pulse repetition rate must remain constant for all NRUN(L).

Units: unitless
Suggested input values: dependent upon the user

NTEST--The number of test exposures run which differ only in pulse repetition rate and/or number of pulses. All other parameters must remain fixed from test exposure to test exposure. This allows reducing computation time if only the pulse repetition rate and/or number of pulses differ from run to run. For single-pulse exposures, NTEST=1.

Units: unitless
Suggested input value: dependent upon the user, MAX = 7

PC--The distance from the pupil to the cornea. PC is required only if the spread function is used (IFIL=1).

Units: cm
Suggested input value: 4.0E-1 for humans
3.6E-1 for rhesus monkeys

POW--The total power per pulse incident on the corneal surface; assumed to be constant during the exposure.

Units: watts
Suggested input value: dependent upon the user

PP--The distance between the pupil and the second principal focal plane. It is required only when the spread function is used (IFIL=1).

Units: cm
Suggested input value: 1.35E-1 for humans
1.2E-1 for rhesus monkeys

PTIME--The uniform time increment into which DPULSE is divided for multiple-pulse calculations. For single-pulse exposures, PTIME is not used.

Units: sec

PTIME = DPULSE/NP for multiple pulses.
NP = 5.

PUPIL--The radius of the pupil of the eye.

Units: cm
Suggested input value: 3.5E-1

PX(L), L=1,LR--The absolute or relative irradiance incident on the cornea for an irregular profile at the radial distance from the center of the beam, RX(L). Symmetry with respect to the axis is assumed. PX(L) cannot have a value of zero at the center axis of the beam.

Units: watts$\cdot$cm^{-2}
Suggested input value: dependent upon the user

QD--All QD values in program RE2, and those associated with the last two TSTEAM values in program RE1, are the power per pulse at the specified grid points required to cause irreversible damage as determined by the damage integral. The other QD values in program RE1 are the powers required to raise the temperature to TSTEAM. QD is set equal to 1.0E+20 when the temperature rise is less than 10^{-3}°C.

Units: watts

QP--The laser intensity at R(1), the center of the beam, entering the eye after the corneal reflection.

Units: cal$\cdot$cm$^{-2}\cdot$sec^{-1}

R(J), J=1,N+1--The radial distance measured from the center of the beam.

Units: cm

R1--The exponential stretching factor in the axial direction for the non-uniform part of the grid system.

Units: unitless

R2--The exponential stretching factor in the radial direction for the non-uniform part of the grid system.

Units: unitless

RCO--The fraction of light reflected from the cornea.

Units: unitless
Suggested input value: Tables A-1 and A-2.

REPET(L), L=1,NTEST--The repetition rate associated with the specific test exposure identified by NRUN(L). All other parameters except the number of pulses must remain constant for all NTEST runs. For a single pulse exposure, set REPET(L)=1. If both NPULSE and NTEST=1, REPET is read but not used in the program.

Units: Hz
Suggested input values: dependent upon the user

RIM--The beam radius at the cornea if the spread function is used (IFIL=1) or at the retina if the spread function is not used (IFIL=0). It is specified at CUT for gaussian profiles (IPROF=1). Although not used for irregular profiles (IPROF=2), a value must always be specified for RIM. For uniform profiles (IPROF=0), it is used with LIM to establish the minimum radial grid increment DR.

Units: cm
Suggested input value: dependent upon the user

RINT--A radial interval used in the input-profile evaluation and in the spread-function integration. It is only printed when the spread function (IFIL=1) or irregular profile (IPROF=2) is used.

Units: cm

RINT = PUPIL/(LI-1), LI=500

RMAX--The maximum radial distance at which damage assessments are to be made. The model assesses damage at all grid points from R(J)=0 to the first grid point beyond R(J)=RMAX.

Units: cm
Suggested input value: 0.001

RPE--A fraction, ranging from zero to one, used to determine the thickness of the two sublayers of the pigment epithelium. RPE is used in conjunction with IGX to determine the absorption properties (APE1 and APE2) of the two sublayers; it cannot equal IGX, thus avoiding a division by zero. RPE represents the fraction of the total thickness of the pigment epithelium (TPE) occupied by the anterior sublayer.

Units: unitless
Suggested input value: 0.--no anterior sublayer
0.33--monkey eye
0.67--human eye
1.0--no posterior sublayer

RPE*TPE = thickness of anterior sublayer
1-RPE*TPE = thickness of posterior sublayer

RRT--The fraction of light reflected from the retina.

Units: unitless
Suggested input value: Tables A-1 and A-2

RSC--The fraction of light reflected from the sclera.

Units: unitless
Suggested input value: Tables A-1 and A-2

RVL--The radial extent of the eye; the boundary where no temperature rise occurs.

Units: cm
Suggested input value: 0.7

RX(L), L=1,LR--The radial distance from the center of the beam, that is associated with the profile irradiance value, PX(L), for irregular beam profiles (IPROF=2).

Units: cm
Suggested input values: dependent upon the user

S--The rate of heat deposition from the incoming beam per unit volume at axial distances Z(I) and radial distances R(J). The S printout is given for N radial positions on one line for each axial grid point except those at the boundaries.

Units: $cal \cdot cm^{-3} \cdot sec^{-1}$

SHB--The specific heat of blood.

Units: $cal \cdot cm^{-3} \cdot {}^{\circ}C^{-1}$
Suggested input value: 0.92

SIGMA--The radius of the beam at the point where the intensity is $1/e^2$ of the maximum value. It is used only for gaussian profiles (IPROF=1) and is specified at the cornea if the spread function is used (IFIL=1). If the spread function is not used (IFIL=0), all profile values are considered to be at the retina.

Units: cm

TAV--The thickness of the ocular media from the cornea to the vitreous humor inclusive; the distance from the cornea to the retina.

Units: cm
Suggested input value: Table A-3

TCH--The thickness of the choroid.

Units: cm
Suggested input value: Table A-3

TIME--The maximum time for temperature rise calculations and damage-integral evaluation.

Units: sec

```
TIME = DT*(XC**KT-1.)/(XC-1.) for single pulse
DT= DPULSE*(XC-1.)/(XC**NP-1.) for single pulse
KT = KTT(L1) for single pulse
NP = NPT(L1) for single pulse
XC = XCT(L1) for single pulse
L1 = ALOG(DPULSE)/.69315 + 29.
TIME = FTIME(L1)*X1 for multiple pulse
X1 = NPULSE/REPET largest fraction in all tests
```

TIMEX(K), K=1,KTYPE--The time at which a plot or selected-time printout of the temperature rises is desired. A separate value of TIMEX(K) must be supplied for each plot or selected-time printout. All values of TIMEX(K) must be less than or equal to the total time over which damage is assessed (TIME).

Units: sec
Suggested input values: dependent upon the user

TO--The initial temperature of the eye.

Units: °C
Suggested input value: 37.

TOM--The transmittance of the ocular media from the anterior surface of the cornea to the pigment epithelium.

Units: unitless
Suggested input value: Tables A-1 and A-2

TPE--The thickness of the pigment epithelium.

Units: cm
Suggested input value: Table A-3

TS(L), L=1,LTMAX,10--The normalized temperature rise decays with respect to time for the melanin granules. They are normalized to the power required to raise a homogeneous pigmented layer an average of 1°C per unit volume and are given in increments of 10BT or 3×10^{-8} sec. Values in Table A-5 were computed for melanin granules 1 μm wide with a 1.5 μm separation between adjacent granules.

Units: °C
Suggested input values: Table A-5

TSC--The thickness of the sclera.

Units: cm
Suggested input value: Table A-3

TSTEAM--A temperature defined by the user according to the particular subject being studied. The model computes the power necessary to raise the temperature of the tissue at specified grid points above the temperature TSTEAM. The model will increment TSTEAM by DTSTM and recompute the required power to exceed the new TSTEAM temperature. TSTEAM continues to be incremented by DTSTM until the power to produce irreversible damage predicted by the damage-integral method is less than the power required to raise the tissue above the temperature TSTEAM. At this point, the power predicted by the damage-integral method is printed. When this occurs twice in sequence, the computation is stopped. This parameter allows the user to determine what powers are necessary to raise the tissue above specified temperatures and to determine the power needed to cause irreversible damage in the tissue.

Units: °C
Suggested input value: 200.

TVL--The thickness of the choriocapillaris.

Units: cm
Suggested input value: Table A-3

VSHX(L), L=1,6--The heat capacity of the Lth ocular media.

Units: $cal \cdot cm^{-3} \cdot {}^{\circ}C^{-1}$
Suggested input value: 1.0

WAVEL--The wavelength of the laser radiation in air.

Units: nm
Suggested input value: 400.-1200. nm

XC--The stretching factor for time intervals associated with temperature calculations.

Units: unitless

XC = XCT(L1) single pulse
L1 = ALOG(DPULSE)/.69315 + 29.
XC = 1.4 multiple pulse

XCT(L1), L1=1,38--An array of expansion factors for calculating time intervals in a single-pulse exposure run.

Units: unitless
Suggested input values: Table A-4

L1 = ALOG(DPULSE)/.69315 + 29.

XFLOW--The rate of blood flow to the tissues surrounding the eye.

Units: $g \cdot cm^{-3} \cdot sec^{-1}$
Suggested input value: .001

XFLOWO(L1), L1=1,6--The total blood flow per unit area leaving the choriocapillaris at a given radial distance.

Units: $g \cdot cm^{-2} \cdot sec^{-1}$

XPD(K), K=1,KT--The normalized temperature rise of the melanin granules at times XT(K). The temperature rises are normalized to the average temperature rise that would occur if the melanin granules were not present. Therefore, if the effects of the melanin granules are not significant, the values for XPD(K) will be 1.0. XPD(K) values are printed in consecutive order for each time that temperature rises are printed.

Units: unitless

XT(K), K=1,KT--The time following the start of an exposure.

Units: sec

Z--In the program output sections Temperature Rises, Predicted Threshold Laser Power, and Radial Extent of Damage, Z is the axial depth from the anterior boundary of the pigment epithelium at which temperature rise and/or damage predictions are printed. Positive and negative numbers indicate axial distances posterior and anterior, respectively,

to the boundary of the vitreous humor and pigment epithelium. In the Grid Information output section, Z is the axial distance from the front of the cornea to the individual grid points.

Units: cm

ZH(I), I=1,M--An axial distance from the cornea to points located halfway between the axial grid points Z(I) and Z(I+1).

Units: cm

ZM--Half the length of the z-axis of the modeled eye.

Units: cm

ZO--The distance of the pupil from the nearest laser beam waist. It must be a positive value; i.e., only diverging beams are applicable. ZO is required only when the spread function is used (IFIL=1).

Units: cm
Suggested input value: 2*RIM/full-angle divergence at RIM, angle in radians.

TABLE A-1. OPTICAL PARAMETERS FOR CAUCASIANS AND NEGROES*

Wavelength nm (WAVEL)	Total transmission from cornea to retina (TOM)	Absorption coefficient (cm^{-1}) for pigment epithelium (APE)		Absorption coefficient (cm^{-1}) for choroid (ACH)		Reflection from cornea (RCO)	Reflection from retina (RRT)	Reflection from sclera (RSC)
		Caucasian	Negro	Caucasian	Negro			
400.0	.094	1838.	1838.	240.	240.	.025	.078	-
500.0	.763	1827.	1355.	106.	203.	.025	.091	.530
514.5	.777	1745.	1261.	99.	195.	.025	.097	.543
520.8	.782	1711.	1223.	97.	194.	.025	.099	.549
530.0	.791	1664.	1170.	93.	192.	.025	.104	.557
600.0	.823	1371.	850.	68.	179.	.025	.133	.620
647.1	.848	1253.	743.	61.	168.	.025	.160	.620
694.3	.853	1144.	643.	54.	159.	.025	.186	.620
700.0	.854	1132.	632.	53.	158.	.025	.189	.620
800.0	.840	974.	433.	43.	136.	.025	.289	.580
900.0	.763	524.	271.	36.	120.	.025	.398	.520
1000.0	.394	373.	205.	32.	109.	.025	.409	.450
1060.0	.492	247.	131.	31.	106.	.025	.450	.426
1064.0	.499	238.	126.	31.	105.	.025	.453	.424
1100.0	.562	164.	79.	30.	103.	.025	.478	.410
1200.0	.082	191.	223.	29.	100.	.025	.394	.320

*From Takata, "Thermal model of laser-induced eye damage."

TABLE A-2. OPTICAL PARAMETERS FOR RHESUS MONKEYS*

Wavelength nm (WAVEL)	Total transmission from cornea to retina (TOM)	Absorption coefficient (cm^{-1}) for pigment epithelium (APE)	Absorption coefficient (cm^{-1}) for choroid (ACH)	Reflection from cornea (RCO)	Reflection from retina (RRT)	Reflection from sclera (RSC)
400.0	.077	1852.	187.	.025	.080	.360
500.0	.826	1545.	169.	.025	.070	.325
514.5	.836	1485.	166.	.025	.070	.318
520.8	.841	1460.	164.	.025	.070	.315
530.0	.847	1425.	163.	.025	.070	.310
600.0	.877	1194.	151.	.025	.070	.265
647.1	.882	1108.	145.	.025	.075	.242
694.3	.887	1028.	141.	.025	.079	.231
700.0	.887	1019.	140.	.025	.080	.230
800.0	(.892)[a]	838.	123.	.025	.095	.250
900.0	(.878)	605.	114.	.025	.144	.245
1000.0	(.790)	434.	110.	.025	.210	.240
1060.0	(.814)	363.	108.	.025	.252	.252
1064.0	(.816)	258.	108.	.025	.255	.253
1100.0	(.830)	313.	107.	.025	.280	.260
1200.0	(.315)	303.	100.	.025	.260	.215

[a]Extrapolated values

*From Takata, "Thermal model of laser-induced eye damage."

TABLE A-3. THICKNESS OF OCULAR MEDIA*

Code	Eye media	Thickness in cm Monkey	Man
TAV	Cornea	$5.16 \cdot 10^{-2}$	$5.86 \cdot 10^{-2}$
	Aqueous humor	$2.9 \cdot 10^{-1}$	$3.1 \cdot 10^{-1}$
	Lens	$3.5 \cdot 10^{-1}$	$3.6 \cdot 10^{-1}$
	Vitreous humor	1.157	1.697
TPE	Pigment epithelium	$1.2 \cdot 10^{-3}$	$1.4 \cdot 10^{-3}$
TVL	Choriocapillaris	$1.0 \cdot 10^{-3}$	$1.2 \cdot 10^{-3}$
TCH	Choroid	$1.68 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$
TSC	Sclera	$1.0 \cdot 10^{-1}$	$1.0 \cdot 10^{-1}$

*From Takata, "Thermal model of laser-induced eye damage."

TABLE A-4. PARAMETERS FOR COMPUTING TIME INTERVALS*

L	NPT(L)	CT(L)	KTT(L)	L	NPT(L)	XCT(L)	KTT(L)
1	1	1.2	47	20	39	1.15	55
2	3	1.2	47	21	40	1.15	56
3	5	1.2	47	22	41	1.15	57
4	7	1.2	47	23	42	1.15	58
5	10	1.2	47	24	43	1.15	59
6	14	1.2	47	25	44	1.15	60
7	18	1.2	48	26	45	1.15	61
8	21	1.2	48	27	46	1.15	62
9	25	1.2	49	28	47	1.15	63
10	28	1.2	49	29	48	1.1	64
11	30	1.2	[illegible]	30	49	1.1	64
12	31	1.2	[illegible]	31	50	1.1	65
13	32	1.2	[illegible]	32	51	1.1	66
14	33	1.2	51	33	52	1.1	67
15	34	1.15	52	34	53	1.1	68
16	35	1.15	52	35	54	1.1	69
17	36	1.15	53	36	55	1.1	69
18	37	1.15	54	37	56	1.1	70
19	38	1.15	54	38	57	1.1	70

*From Takata, "Thermal model of laser-induced eye damage."

TABLE A-5. NORMALIZED TEMPERATURE RISE DECAYS*

L	TS(L)									
1-91	8.000	7.359	6.815	6.349	5.944	5.590	5.277	4.999	4.749	4.523
101-191	4.319	4.132	3.960	3.802	3.656	3.520	3.394	3.276	3.166	3.062
201-291	2.965	2.874	2.787	2.706	2.629	2.557	2.488	2.423	2.361	2.303
301-391	2.247	2.194	2.144	2.096	2.051	2.007	1.966	1.927	1.889	1.853
.	1.819	1.786	1.755	1.725	1.697	1.670	1.644	1.619	1.595	1.572
.	1.550	1.529	1.509	1.490	1.472	1.454	1.437	1.421	1.405	1.390
.	1.376	1.362	1.349	1.336	1.324	1.312	1.301	1.290	1.280	1.270
.	1.260	1.251	1.242	1.233	1.225	1.217	1.209	1.202	1.195	1.188
.	1.181	1.175	1.169	1.163	1.157	1.152	1.147	1.141	1.137	1.132
1001-1091	1.127	1.123	1.119	1.115	1.111	1.107	1.103	1.100	1.096	1.093
.	1.090	1.087	1.084	1.081	1.078	1.075	1.073	1.070	1.068	1.066
.	1.063	1.061	1.059	1.057	1.055	1.053	1.052	1.050	1.048	1.047
.	1.045	1.043	1.042	1.041	1.039	1.038	1.037	1.035	1.034	1.033
.	1.032	1.031	1.030	1.029	1.028	1.027	1.026	1.025	1.024	1.024
.	1.023	1.022	1.021	1.021	1.020	1.019	1.019	1.018	1.017	1.017
.	1.016	1.016	1.015	1.015	1.014	1.014	1.013	1.013	1.012	1.012
.	1.012	1.011	1.011	1.010	1.010	1.010	1.009	1.009	1.009	1.008
.	1.008	1.008	1.008	1.007	1.007	1.007	1.007	1.006	1.006	1.006
.	1.006	1.006	1.005	1.005	1.005	1.005	1.005	1.005	1.004	1.004
.	1.004	1.004	1.004	1.004	1.004	1.004	1.003	1.003	1.003	1.003
2001-2091	1.002	1.002	1.002	1.002	1.002	1.002	1.002	1.002	1.002	1.002
2101-2191	1.002	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001

*From Takata, "Thermal model of laser-induced eye damage."

TABLE A-6. ZERO-ORDER BESSEL FUNCTION*

L	Zero Order Bessel Fn JO(L)	L	Zero Order Bessel Fn JO(L)
1	1.00000	17	.45540
2	.99750	18	.39798
3	.99002	19	.33998
4	.97762	20	.28181
5	.96039	21	.22389
6	.93846	22	.16660
7	.91200	23	.11036
8	.88120	24	.05553
9	.84628	25	.00250
10	.80752	26	-.04838
11	.76519	27	-.09680
12	.71962	28	-.14244
13	.67113	29	-.18503
14	.62008	30	-.22431
15	.56685	31	-.26005
16	.51182	32	-.29206

*From Takata, "Thermal Model of laser-induced eye damage."

TABLE A-7. REFRACTIVE INDEXES*

L	Wavelength nm	Refractive index NA(L) (WATER)	L	Wavelength nm	Refractive index NA(L) (WATER)
1	350	1.357 (not water)	12	900	1.328
2	400	1.346 (not water)	13	950	1.327
3	450	1.341 (not water)	14	1000	1.326
4	500	1.336	15	1050	1.325
5	550	1.334	16	1100	1.324
6	600	1.332	17	1150	1.3235
7	650	1.331	18	1200	1.323
8	700	1.330	19	1250	1.322
9	750	1.329	20	1300	1.321
10	800	1.328	21	1350	1.320
11	850	1.327	22	1400	1.320

*From Takata, "Thermal Model of laser-induced eye damage."

APPENDIX B

PLOT ROUTINE

The plot routine was developed to display two- and three-dimensional temperature rise profiles as a function of radial and axial coordinates at selected times. At specified grid points, the routine utilizes card-punched temperature data that are output by the retinal program, and will generate, for each data set, as many plots as desired. For each plot, the user specifies the physical size of the plot area on the Model 1765 Calcomp plotter and can view the profiles at any angle desired by using a succession of rotation, scaling, and translation commands.

The R-axis, Z-axis, and T-axis of the temperature rise plots refer to the radial coordinates, the axial coordinates, and the temperature rises, respectively. The <u>permanent</u> viewing axes (x,y,z) are set up in a right-hand coordinate system with the permanent x-axis horizontal to the right, the y-axis vertical and up, and the z-axis coming perpendicularly out of the viewing plane. Initially the RZT axes and the permanent xyz axes have the same orientation and origin. All rotations and translations are in relation to the <u>permanent</u> axes and <u>independent</u> of any previous commands. Good three-dimensional views are obtained by a succession of these commands.

The input deck for the plot routine for a single set of temperature data can be separated into two sections. The first section consists of the necessary temperature data on cards that are punched as output from the retinal program. The data cards, for each selected time, should be placed as input to the plot routine in the order in which they are punched with one exception. After the retinal program punches the cards containing the temperature data for the selected times from any one run, it punches MAX RGV CARD(S) FOLLOW on a comment card. This is followed by a number of cards, equal to the number of selected times (KTYPE) and each containing the maximum temperature rise. This maximum rise is used to determine a scaling factor for the temperature rises. The scaling factor is a power of 10 chosen internally to put the maximum temperature rise in the 1-to-12 range. The comment card (MAX RGV CARD(S) FOLLOW) should be discarded. A maximum-temperature-rise card must be placed at the end of the set of temperature data for each selected time. When cards have been punched for more than a single selected time (KTYPE > 1), the end of each set of temperature data can be found by locating the initial card of the succeeding set. This initial card contains NRUN, NPULSE, and REPET and is the only card with the format 2I7, E10.4. When only one selected time (KTYPE=1) has been punched, the only deck manipulation is to discard the comment card.

The second section contains the command data for plotting. These commands scale, rotate, and translate the axes and establish the viewing screen for the desired plots. A uniform format for all commands and their associated parameters is used:

Columns 1-4	Columns 11-20	Columns 21-30	Columns 31-40
keyword left-justified	1st parameter	2d parameter	3d parameter,etc.

To identify the desired command, a keyword in alphanumeric format is entered in columns 1-4. Parameters, as applicable, are entered as floating-point numbers in 10-character-wide fields starting with column 11. A blank entry is always read as a floating number with value zero.

Several commands normally precede any others when the sequence of input plotting commands is set up. The first command, DUM, has no effect on the actual plot setup, but requests a summary of all the points in the data base--with the low, high, and mean values for the R, Z, and T ranges of data to be printed.

The second command, SCRN, sets up the size and position of the display area as measured on the Calcomp plotter. Without this command, the program will not plot.

The third command in the sequence, BOX, scales the object to fit the viewing area established by SCRN and centers the object on the origin. This eliminates losing plots due to disparity between coordinate magnitudes.

Without any further information, the program would plot an isometric R-Z view of the object, giving a plot of the radial vs axial grid points. Table B-1 contains a sample input deck used to obtain an R-T view (radial vs temperature), a Z-T view (axial vs temperature), and a good three-dimensional isometric view. Cards 1 through 17 contain the data and information supplied by the Retinal Thermal Model. Within this section, cards 11 through 16 contain the actual temperature data to be plotted. Card 17 is the maximum RGV value card. Cards 18 through 33 contain the individual plot commands. These can be used with any set of input data to obtain the same basic results. The plots generated by these commands are shown in Figures B-1 through B-3.

The three rotation commands, PITC, YAW, and ROLL (about the permanent xyz axes) are the most commonly used commands to move the object and obtain the desired view. The command TRAN can also be used to move the object through a translation relative to the permanent origin.

In addition to the positional commands, several commands can be used to scale the temperature rise profiles and change the viewing perspective. Two commands (besides BOX) have a scaling effect on the plots: SCAL can rescale the R, Z, and T coordinates independently; and FACT simply blows up or shrinks all plotting by applying the same scaling factor to all three of the axes. The command DIST, used to determine the viewing perspective of the plotted object, allows the viewer to adjust his position and distance relative to the permanent origin and to specify his distance

from the plane onto which the three-dimensional object has been projected. Without DIST, the program assumes an isometric view, with the permanent origin lying in the projection plane. If an enlargement of a portion of a plot is desired, the WIND command is used. This automatically scales up the area of interest to fill the screen, and the rest of the plot is cut off.

To obtain any plotting, the command PLOT must be used. This calls on the plot subroutine to plot the current view of the object as defined by the previously built-up commands. Normally, the plot includes all lines whether or not they would be seen by the observer of the three-dimensional object. The hidden lines can be dashed or totally removed by using the command HIDE. The visibility of a line is determined by the surface normal vectors entered in the plot file, which can be reversed by the command SIGN. After a plot command, a quick reinitialization of the transformation matrix is achieved by the command REIN. This erases all of the previously built-up results from the positional and the scaling commands.

The sequence of plotting commands listed in Table B-1 is generally adequate for plotting temperature profiles; however, the commands and their sequence can be changed at any time to fit the user's desire. The list in Table B-1 is given as a description of each input command and its associated parameters and is presented in sequence of general usage.

<u>DUM command</u>--Requests a summary of the current number of points in the data base and of the R, Z, and T ranges of the data. DUM is usually the first command entered in any command sequence and has no parameters associated with it.

<u>SCRN(A B C D E) command</u>--Sets up the physical size of the display area and draws a border around that area for every plot. Without either this or the window command, the Calcomp plotter will not plot. For each set of temperature data, the screen command remains in effect and is affected only by subsequent screen commands. Parameters A and B are the coordinates of the lower left corner of the screen in reference to the permanent origin; C and D describe width and height; and E, the fifth parameter, may be entered to define a three-dimensional rectangular box with E as the depth (units are all in inches).

<u>BOX(A B C) command</u>--Causes the object being plotted to fill a fraction of the screen area. The object is first moved so that its center of gravity is coincident with the permanent origin and then rescaled from there to fill a proportion of the available viewing area, as determined by parameters A, B, and C. When only A is entered, a single scale is applied to all three axes. If all three parameters are entered, the object is scaled to fit the A, B, and C proportions of the specified x, y, and z screen dimensions respectively. The parameters are generally set at values slightly less than unity; such as 0.9 or 0.85.

ROLL(A) command--Indicates that the object should move counterclockwise in the viewing plane by an angle of A degrees. The permanent z-axis is the axis of rotation.

PITC(A) command--Indicates that the object should rotate A degrees around the permanent x-axis so that the top part of the screen will come toward the viewer.

YAW(A) command--Indicates that the object should rotate A degrees around the permanent y- (or vertical) axis so that the rightmost portion of the screen will move away from the viewer.

PLOT(A B) command--Causes the current view (as defined by BOX, DIST, ROLL, PITC, YAW) of the object to be plotted. Parameters A and B define the relative X and Y advance on the Calcomp plotter for a permanent new origin of coordinates. A and B are interpreted as real inches. So that the title of the plots and scaling information will be appropriately displayed for each set of plotting data, A=12.75 must be on the first PLOT command card used. Also, B=0 must be on every PLOT command card after the first so that succeeding plots have a common baseline.

DIST(A B X Y) command--Adjusts the distance of the observer from the object. If no parameter or zero-valued parameters are entered, the view will be isometric. If both A and B are nonzero, A is the distance of the viewer from the projection plane and B is the distance of the viewer from the permanent origin. When A is nonzero and B is zero, parameter A is applied to both distances. Optional third and fourth parameters, X and Y, may be added to allow the viewer to shift his viewing position with respect to the z-axis. (All four parameters are in units of inches.)

REIN command--Reinitializes the object to its original position by unitizing the transformation matrix. All previously built-up results from roll, pitch, yaw, scale, box, and translation commands are lost.

HIDE(A) command--Calls for a change in the use of the hidden-line calculation. Through this calculation, lines not normally seen by an observer of a three-dimensional object may be dashed or removed. If parameter A is zero, the hidden-line calculation is not used and all lines are drawn. If A is 1.0, the hidden lines are removed; and if A is 2.0, the hidden lines are dashed. The most recent HIDE command will remain in effect until it is superseded by another HIDE command.

FACT(A) command--Simply expands or shrinks all plotting dimensions along all axes by factor A.

SCAL(A B C) command--Rescales the current object. If factors B and C are both zero, all three dimensions are rescaled uniformly by factor A. In this situation, the commands SCAL and FACT are identical. Otherwise, the R, Z, and T coordinates are independently scaled by factors A, B, and C respectively.

SIGN (A) command--Used to reverse the sense of the surface normals entered in the data base. To do this, A should be set equal to -1.0.

TRAN (A B C) command--Effects a translation of the current object position through a vector (A,B,C) relative to the permanent origin. A, B, and C are in terms of inches along the permanent x, y, and z axes, respectively.

WIND (A B C D) command--Used to zoom in on any portion of the current plot. A and B are the lower left-hand coordinates of the windowed area, and C and D give the horizontal and vertical extent of the windowed area in terms of the permanent display coordinates. The windowed area is then blown up to fill the entire screen area. If the screen command has not been effected, this command acts as a screen with A, B, C, and D having the same meaning as their equivalents for SCRN. The window command is only in effect for the immediately following plot, but can be reactivated by entering a WIND card with no parameters. In this case, the previous window, with its parameters, is put into effect.

TABLE B-1. SAMPLE PLOT INPUT (WITH COMMENTS)

Information and Data from Retinal Model

```
 1.         3           1          1.0
 2.     .200E-07     .530E 03    .200E 00
 3.         9          14          10          1         6
 4.        11          29
 5.    0.0        0.0003      0.0007      0.0010      0.0013      0.0030      0.0109      0.0496      0.2387      1.1613
 6.    5.6637
 7.   -0.9575     0.9575      1.5664      1.7599      1.8215      1.8411      1.8473      1.8493      1.8499      1.8501
 8.    1.8503     1.8505      1.8507      1.8509      1.8511      1.8513      1.8515      1.8517      1.8519      1.8521
 9.    1.8523     1.8529      1.8549      1.8611      1.8807      1.9423      2.1358      2.7447      4.6597
10.     .200E-07
11.    0.840000E-03    0.640000E-03    0.330000E-03    0.160000E-03    0.900000E-04    0.100000E-04
12.    0.461873E 01    0.348269E 01    0.178623E 01    0.893650E 00    0.490790E 00    0.545600E-01
13.    0.209787E 01    0.158187E 01    0.811320E 00    0.405900E 00    0.222920E 00    0.247800E-01
14.    0.565800E-01    0.426600E-01    0.218800E-01    0.109500E-01    0.601000E-02    0.670000E-03
15.    0.541700E-01    0.408400E-01    0.209500E-01    0.104800E-01    0.576000E-02    0.640000E-03
16.    0.524300E-01    0.395300E-01    0.202800E-01    0.101400E-01    0.557000E-02    0.620000E-03
17.    0.461873E 01
```

Command Data for Plotting

```
18.  DUM                                                          (Dump summary of all 3-D points)
19.  SCRN      -4.1      -3.1      8.      6.      6.       (Establish screen size and position)
20.  BOX        1.0       1.0      1.0                             (Rescale object to fill screen)
21.  PITC     -90.                                             (Pitch object -90° for an X-Z view)
22.  BOX        0.9       0.9                  (Scale view to fill 0.9 of X and Y dimensions of screen)
23.  PLOT      12.5       6.          (Advance 12.5 in (31.8 cm) in X, 6 in (15 cm) up in Y, and plot)
24.  YAW      -90.                                                  (Yaw by -90° to give Y-Z view)
25.  BOX        0.9       0.9                  (Scale view to fill 0.9 of X and Y dimensions of screen)
26.  PLOT      10.        0.                                  (Advance 10 in (25 cm) in X, and plot)
27.  REIN                                                (Reinitialize to give an isometric X-Y view)
28.  BOX        1.0       1.0      1.0                               (Scale object to fill screen)
29.  PITC     -90.                                                             (Pitch object -90°)
30.  YAW      -45.                                                               (Yaw object -45°)
31.  PITC      30.                                                              (Pitch object 30°)
32.  BOX        0.9       0.9      0.9                          (Rescale view to fill 0.9 of screen)
33.  PLOT      10.        0.                                  (Advance 10 in (25 cm) in X, and plot)
```

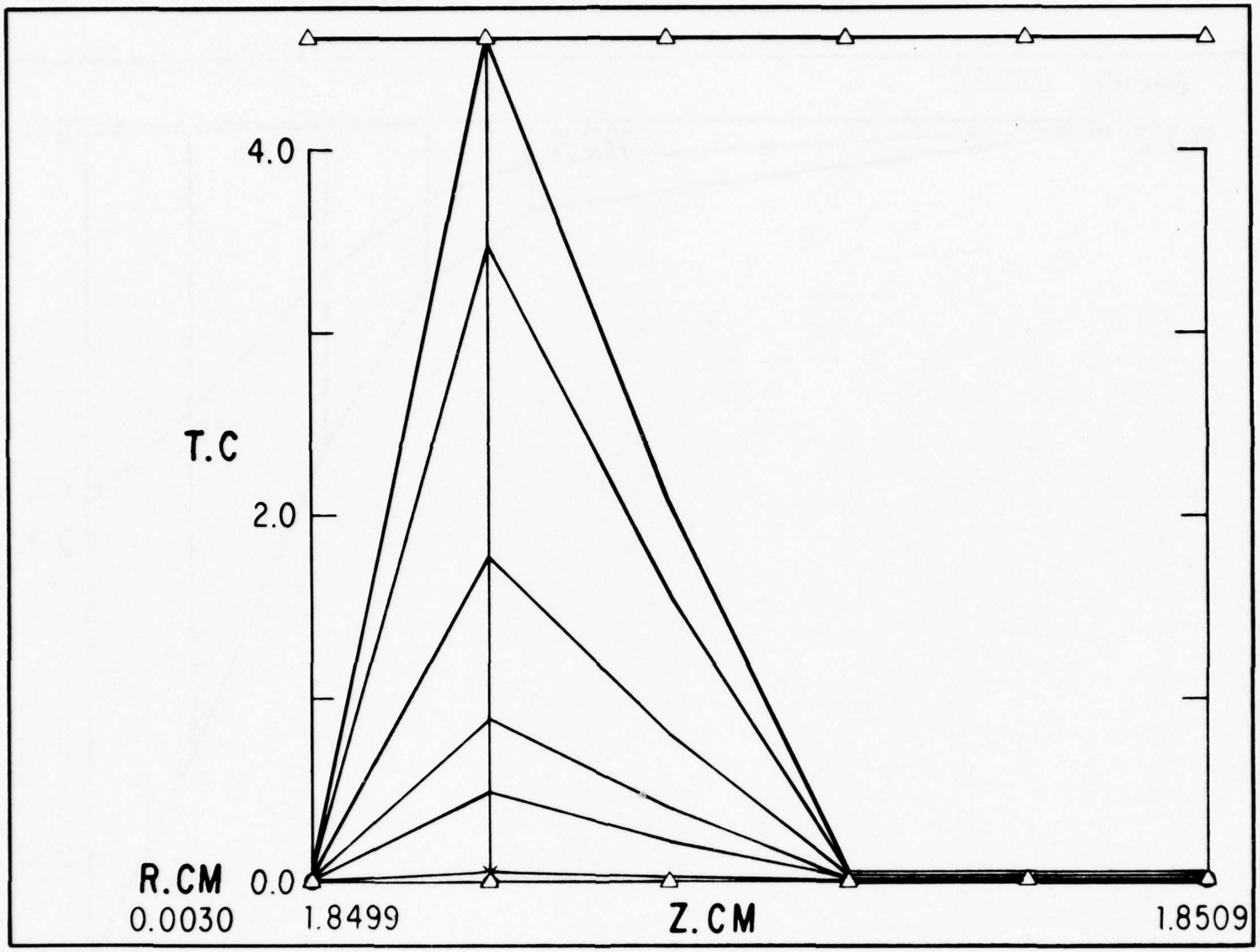


Figure B-1. Temperature rise versus axial depth.

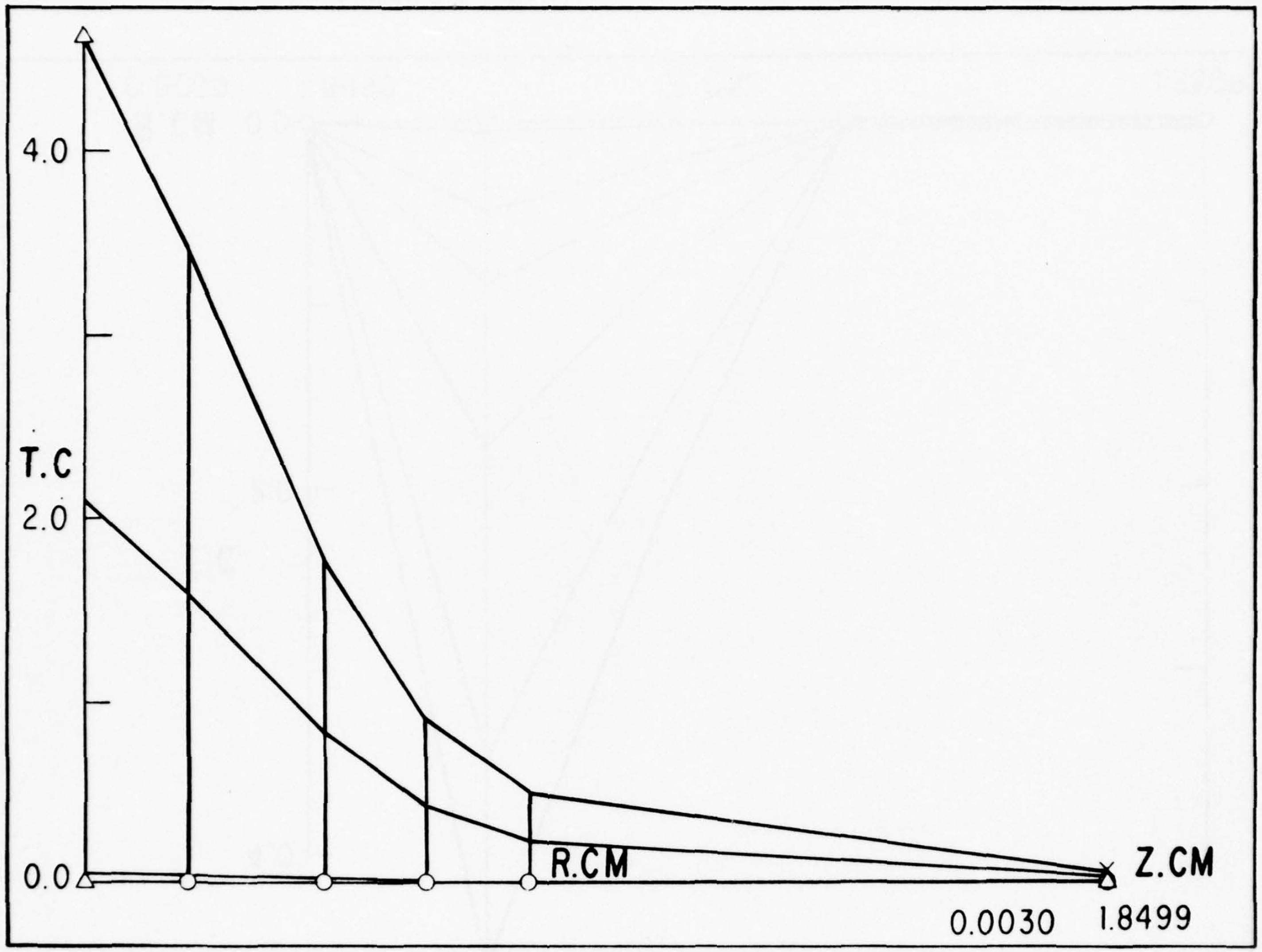


Figure B-2. Temperature rise versus radial extent.

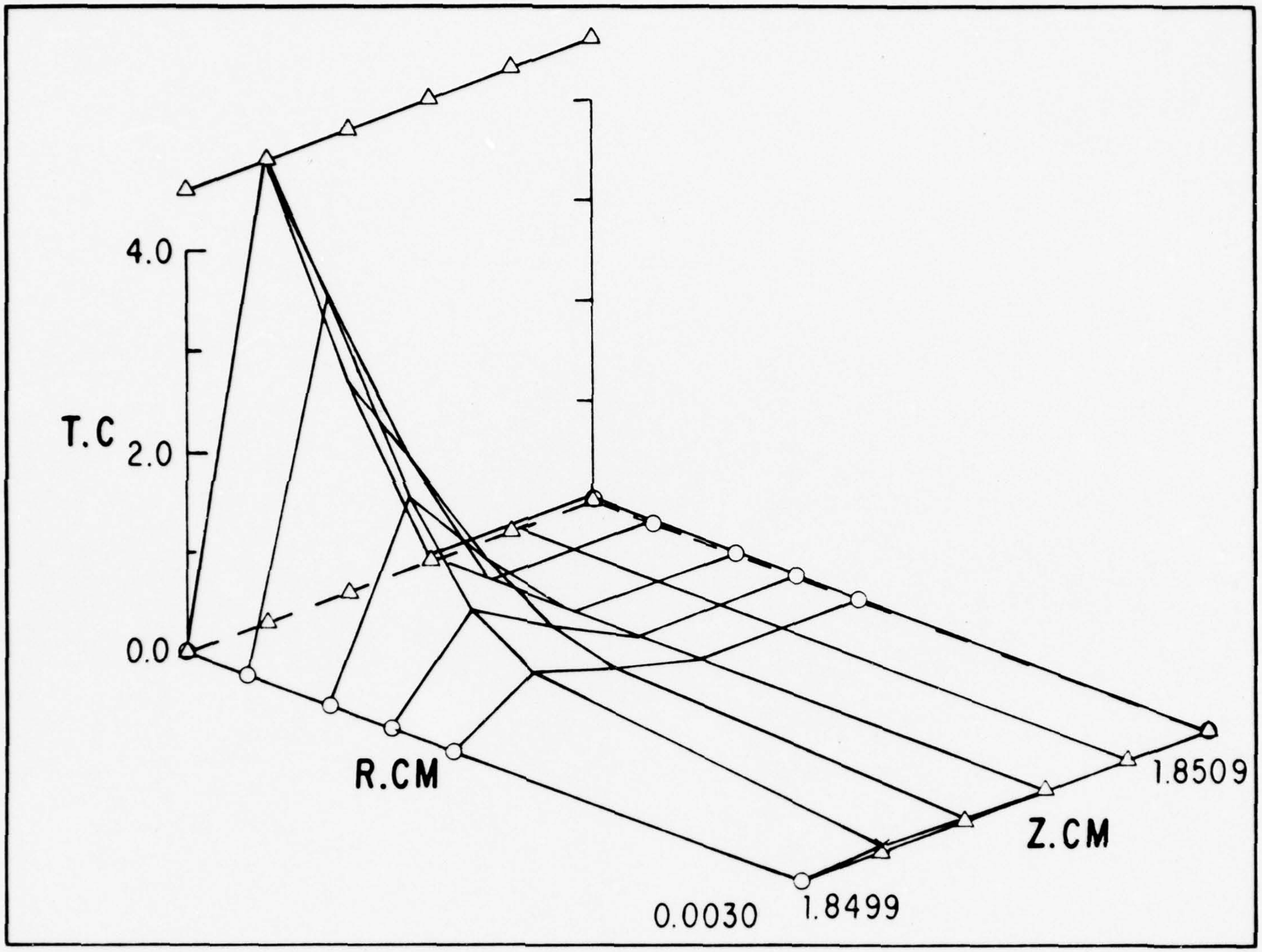


Figure B-3. Three-dimensional view of temperature rise.

PRECEDING PAGE BLANK-NOT FILMED

APPENDIX C

INPUT-OUTPUT PROCESSES

This appendix will provide the user at Brooks AFB with the basic information necessary to run the retinal model on an IBM 360/65 computer, and will serve as an example for similar setups. One such computer is located at the San Antonio Data Service Center (SADSC), with a remote terminal at Brooks AFB. The prospective user should be familiar with the required input data cards as outlined in text, Input Requirements section.

At Brooks AFB, the retinal models (HBRØ1RE1 and HBRØ1RE2) are stored on a computer disk library. This eliminates having to submit and recompile the entire program for each set of data cards. Therefore, in addition to the data cards, only the Job Control Language (JCL) cards are necessary. These are used to call the stored program and to set up the program for operation on the IBM 360/65. Figure C-1 is an example of a deck used to call up a stored program.

Of the JCL cards, the job (JOB), execute (EXEC), and data definition (DD) cards are required by any IBM 360 operating system. The SETUP card is required by SADSC for long-running programs and programs requiring large core. In addition, several of the parameters on the JOB, EXEC, and DD cards are controlled by SADSC. The following is a list of these JCL cards with the parameters and formats required to call and run the retinal program (HBRØ1RE2) on the computer.

(1) JOB. The job cards identify the beginning of a new job; therefore, they must always be the initial cards in the deck setup. They are variable-field control cards, but have certain requirements placed on them by SADSC. They should fit the following format:

```
//HBaaabbbØJOBØ(3HØ1,BØ2Ø,cccc,ddd,eeee,,,Y,ff),'HBggggg1ØRØhhhhhhhh',
//ØCLASS=H,PRTY=5,MSGCLASS=A,MSGLEVEL=(2,Ø)
```

The parameters that are variable, depending on the user and the job being run, are represented by the lowercase letters above and are as follows (Ø must be blank):

aaa - Unique user code assigned to each user for identity.

bbb - Up to 3 alphanumeric characters (plus @, #, and $ when desired) assigned by the user to identify the job. This and the user code, together, make up the job name.

cccc - Job execution time in minutes. This is the total estimated time for job execution--the sum of the central processing unit (CPU) time, wait time, and input/output (I/O) time requirements--and may consist of up to 4 digits. Details on the SADSC job class requirements (set forth later in the sequel) will help the user arrive at a suitable time estimate. A suggested time estimate is 1.5 times the amount of CPU time entered on the EXEC card.

ddd - Estimated output line count (in thousands of lines), consisting of up to 3 digits. It is recommended that this value be set at 9 and changed as experience dictates. SADSC operators will automatically cancel the job if the specified line count is exceeded by 9000.

eeee - Estimated card count (in hundreds of plot data cards to be punched), consisting of up to 4 digits. A card count based on an average of 30-40 plot cards for each set of temperature rise values corresponding to a selected time is recommended. SADSC operators will cancel the job if the card count is exceeded by 3000.

ff - Maximum number of lines to be printed per page. This may consist of up to 2 digits, up to a value of 61. To fully utilize the output paper, the value of 61 is recommended.

gggggg - Cost-accounting code associated with the particular work unit under which the job is being run. If this code is less than 6 characters, it must be left-justified, with @ signs acting as fill characters on the right to complete the 6-character subfield.

hhhhhhhh - User's last name. Up to 8 alphabetic characters may be used.

The cost-accounting code is the only item requiring its associated subfield to be complete. When not used to the maximum, other subfields should be closed up to include only that portion being used. All other parameters and values on the JOB cards should be included and left as they are. However, the time requirements may necessitate a change in "CLASS=" designation as set forth later in the sequence.

(2) COMMENT. Cards having //* in the first three columns may be used as comment cards to supply information concerning the program to the user. They should be placed after the JOB cards but before any data definition cards. They have no effect upon the running of the program.

(3) SETUP. Special resources required during job execution are indicated by the setup card. It is listed on the computer console when the job enters the system, alerting the SADSC computer operator of any requirements for large amounts of CPU time and/or core storage. For retinal model RE2, the format for this card is:

```
/*SETUPØØØØØØØØ'376K CORE ROD, aaa CPU MINS'
```

Columns 8-15 on the card should always be blank, and the message must be in quotes. The number of CPU minutes depends on the number of data sets being run and the data itself. A trial value of 1 minute for each NRUN set of data is suggested. For 4 sets of data, the CPU time would be 4. If CPU time limit exceeds 10 minutes, the job should be submitted as two or more jobs.

(4) EXEC. The execute card tells the computer what type of action the user wants on the source or data deck which will follow. For RE2, the EXEC card has the following format:

```
//STEP1ɓEXECɓFORTGO,PROGRAM=HBRØ1RE2,REGION.GO=376K,TIME.GO=aaa
```

This card identifies the GO step as the step to be executed. The GO step calls for execution of the program named HBRØ1RE2, which has been compiled in FORTRAN IV language. This card further requests a core size of 376K and sets a CPU time limit (aaa) on the execution of the GO step. This time limit should equal the time requirement quoted in the SETUP message. If either the core size or the CPU time limit request is exceeded, program execution will be terminated.

(5) DD. The data definition cards basically supply the computer with descriptions of data sets. Two such cards are required in core loading and running the RE2 program. They are as follows:

```
//STEPLIBɓDDɓDSN=SYS1.TESTLIB,DISP=SHR
//GO.SYSINɓDDɓ*
```

The first card identifies the system library (TESTLIB) in which the program mentioned in the EXEC statement is stored. The second card identifies the cards which follow it as data cards for the GO step.

(6) DELIMITER (/*). A card with /* in the first two columns (referred to as a delimiter card) must follow the data card deck. It serves as the end-of-file card for the card deck.

Occasionally, the user may need to recompile the program (RE2) and restore it in the computer library. To do this with a data run would require a deck setup as in Figure C-2. The JOB, SETUP, EXEC, and DD cards require some changes and additions:

(1) JOB. The only change required in the job card for compiling and running the RE2 program is in the estimated job execution time. The usual total estimated job time should be increased by 2 minutes in order to satisfy compiler and linkage editor time requirements.

(2) SETUP. As on the JOB card, the addition of compiler and linkage editor time requirements necessitates an increase in the quoted CPU time requirement. The usual time requirement quoted on the SETUP message for running from the disk library should be increased by 2 minutes to satisfy the extra time requirement.

(3) EXEC. To compile and run the RE2 program, the execute card has the following format:

```
//STEP1ƀEXECƀFTG1CLG,REGION.FORT=164K,REGION.LKED=114K,REGION.GO=376K,
//ƀTIME=aaa
```

This card identifies FORT (FORTRAN), LKED (linkage editor), and GO (execution) as steps to be executed. The FORT step compiles the program, the LKED step edits and stores the program, and the GO step executes the program. The card requests core sizes of 164K for FORT, 114K for LKED, and 376K for GO; and sets a CPU time limit (aaa) to accomplish steps FORT, LKED, and GO. This time limit should be equal to the time requirement quoted on the SETUP message.

(4) DD. Compiling, storing, and running the RE2 program requires three data definition cards. They are formatted as follows:

```
//FORT.SYSINƀDDƀ*
//LKED.SYSLMODƀDDƀDSN=SYS1.TESTLIB(HBRØ1RE2),DISP=SHR
//GO.SYSINƀDDƀ*
```

The first DD card identifies the cards that follow it as source cards for the FORT step. A delimiter card follows the source, or program, deck. Immediately after the source-deck delimiter card, the second DD card directs the computer to store the program in system library TESTLIB under the name HBRØ1RE2. The last DD card identifies the cards that follow it as data cards for the GO step. A delimiter card is at the end of the data deck.

The SADSC IBM 360/65 computer system has a scanning procedure in operation to detect JCL card error. Detection of a single JCL error by the scanner terminates further processing of the job. One such error detected is a job class error. Job class is determined by use of core requirements and CPU characteristics. Specifically, the ratio of estimated job time (on JOB card) to the time request entered on the EXEC card is considered as $\geq 2:1$ or $<2:1$, and the job is considered I/O bound or CPU bound according to these ratio values. The user selects the proper job class by using the following table of job class requirements:

Core requirements	$\geq 2:1$	$<2:1$	Job Classes
Max Region $\leq$ 74K (DEFAULT)	A	B	
75K $\leq$ Max region $\leq$ 150K	C	D	
151K $\leq$ Max region $\leq$ 300K	E	F	
301K $\leq$ Max region	G	H	
Special classes not verified	O,N,T,J	O,N,T,J	

If the user wants to run the RE1 program, which uses the MXGRAN subroutine, the following changes must be made:

(1) The name of the program changed from HBRØ1RE2 to HBRØ1RE1.

(2) The core requirement for execution (GO) increased from 376K to 436K on both the SETUP and the EXEC cards. All other parameters would be used as outlined for the RE2 program.

The plot routine is handled in the same manner as the main retinal program and is stored in the computer library; therefore, it has the same basic JCL card setup as has been outlined for the retinal program. An example of a deck used to call and run the plot routine is shown in Figure C-3. The changes that are required are:

(1) The name of the program is HBRØ1PLT.

(2) The core required by the GO step is 148K. This change should be reflected on the SETUP and EXEC cards.

(3) For normal running, FORTGO on the EXEC card should be replaced by PLOTGO. For compiling, the equivalent of FTG1 on the EXEC card is PLOTG, and the core request for FORT should be REGION.FORT=12ØK.

(4) The TIME.GO entry on the EXEC card should be approximately 0.05 times the number of plots desired.

(5) Set both estimated time and line count to 10 on the job card, and adjust as experience dictates. The number of cards to be punched should be set to zero.

(6) The ratio of estimated job time (on JOB card) to the time request entered on the EXEC card must be evaluated to determine the proper job class as outlined above. This is controlled by the parameter "CLASS=" on the JOB card.

(7) A delimiter card goes at the end of each set of data.

(8) For a single data set, a DD card (//GO.FTØ5FØØ2ɃDDɃ*) must follow the data-set delimiter card and, in turn, must be followed by a delimiter card. For multiple sets of data to be run for any given job, a DD card having the following format must precede each data set subsequent to the first set:

//GO.FTØ5FaaaɃDDɃ*

A 3-digit number (aaa) indexes the sets of data in sequential fashion; for example, aaa=ØØ2 for the second set of data [TIMEX(2)], aaa=ØØ3 for the third set [TIMEX(3)], etc. A DD card of this format must also follow the last data-set delimiter card and must have the proper index number for an additional data set, but with a delimiter card following it.

The computer terminal and its related facilities at Brooks AFB are controlled by Biometrics Division of the USAF School of Aerospace Medicine.

Before running any jobs, the user should familiarize himself with the area where card decks are submitted and returned and output is picked up. Two tables serve these purposes. Decks to be run are placed in the tray on the input table. Also on the input table is a log sheet on which the user must record each job submitted, and small punch/plot cards which must be filled out and placed with the card deck whenever punched cards or plots are expected as output.

All output, whether printed, plotted, or punched, is placed on an output table. Card decks which have been run are placed in trays on this table, with each tray filed corresponding to a range of user-code initial characters. The space on this table is limited, so users should pick up their card decks and output within a reasonable time.

A requirement for using the computer is having a valid user code. A prospective user can get a user code from the director of the programmers, who can also help in identifying or setting up the proper cost-accounting codes assigned to different work units. If either of these codes are invalid on the JOB card, the job will not run.

Several computer-terminal operators are constantly in the input/output area. Questions regarding any part of the input/output process and requests for assistance with any of the machines may be directed to these operators.

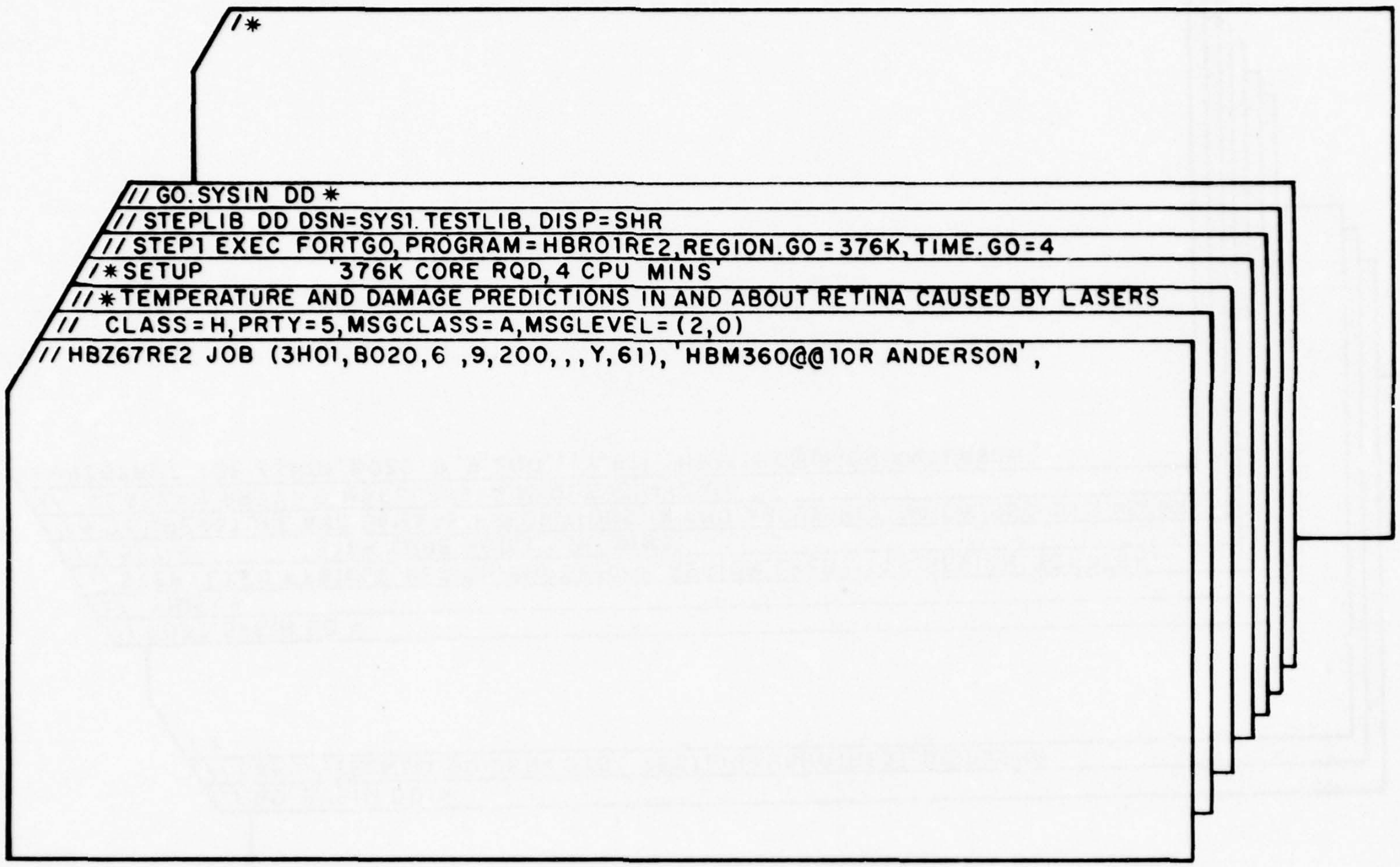


Figure C-1. Sample card deck for running retinal program RE2.

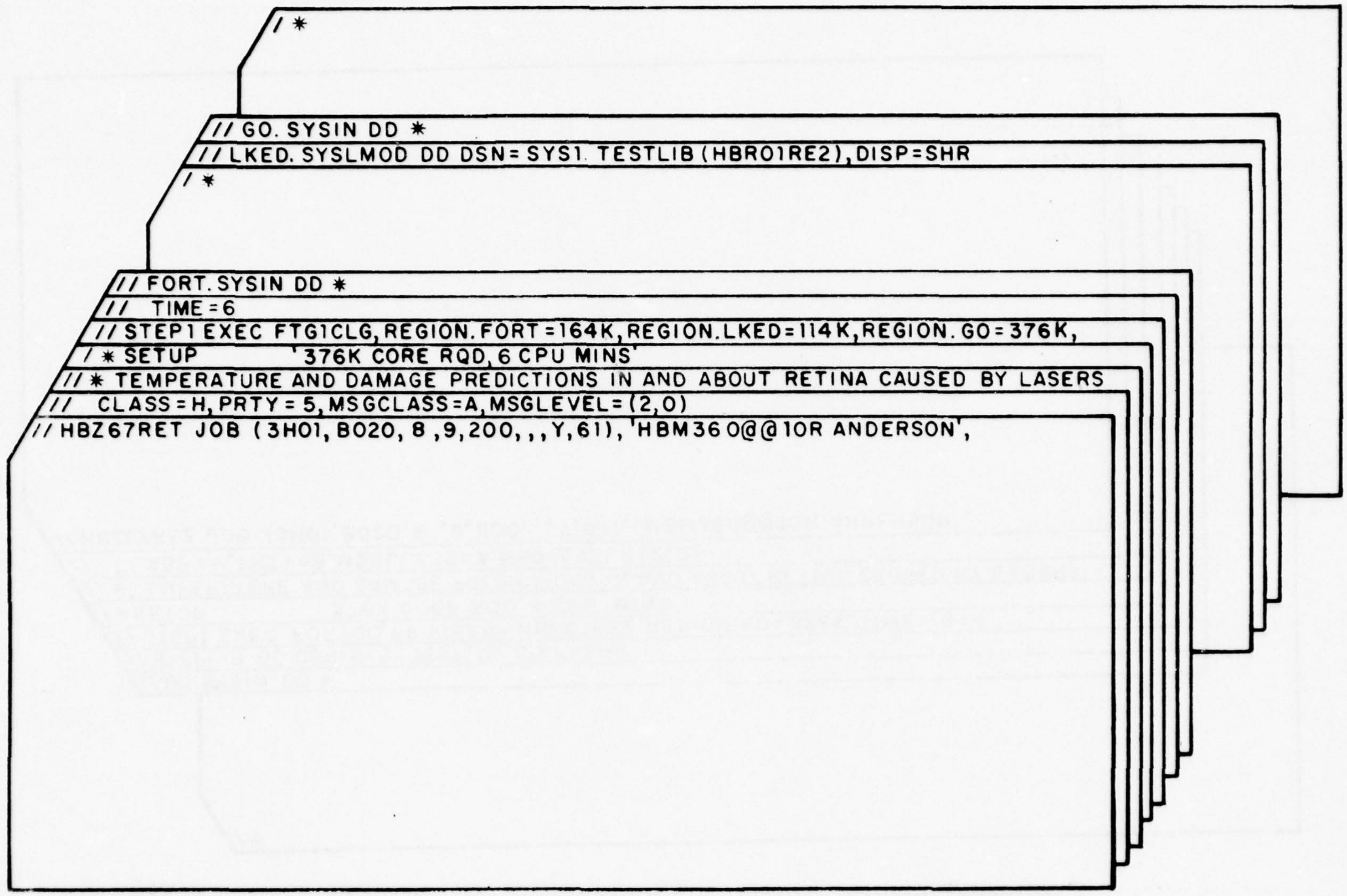


Figure C-2. Sample card deck for compiling and running retinal program RE2.

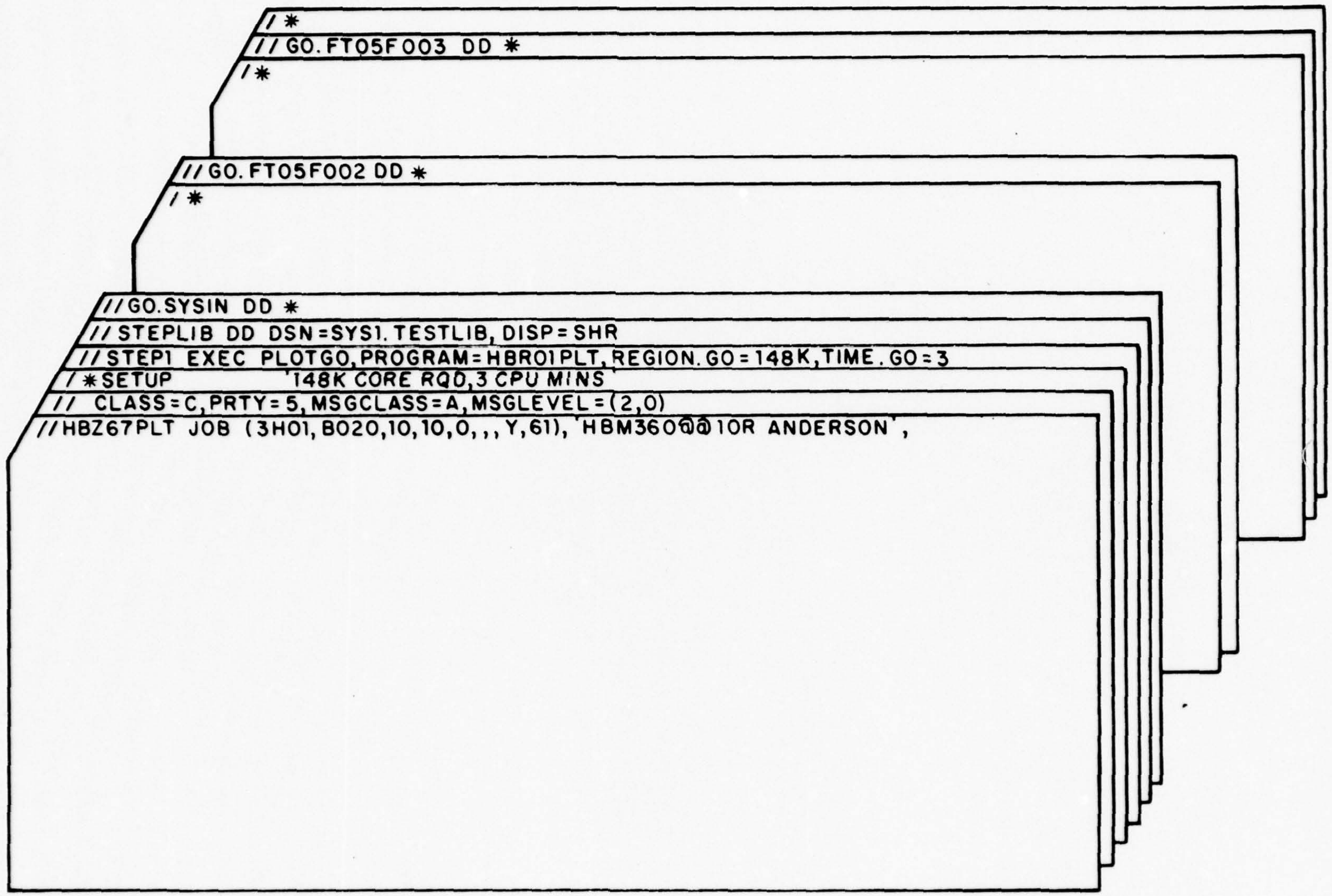


Figure C-3. Sample card deck for running the plot routine.

PRECEDING PAGE BLANK-NOT FILMED

APPENDIX D

PROGRAM LISTING

```
C                        RETINAL MODEL   IITRI
C                        VERSION 1  14 NOV 1975
C   TEMPERATURE AND DAMAGE PREDICTIONS IN AND ABOUT RETINA CAUSED BY LASE
C      NEW PRINTOUT TITLES AND GROUPINGS OF INFORMATION
C      UTILIZES SUBROUTINE MXGRAN
C
      COMMON A(29,3),AP,AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),
     2IAB(29,14),IBLOOD(10),IFIL,IG,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2,
     4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RPT,
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPR,TVL,TS(2200),TSC,TTS,V(29,14)
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),
     7XPD(120),XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC
      DIMENSION CXC(14),CXR(29),DAMAGE(2,2),DXC(14),DXR(29),FTIME(38),
     1FXC(14),FXR(29),ID(230),JD(230),KTT(38),NPT(38),NPULSE(7),NRUN(7),
     2QD(29,14),REPET(7),TIMEX(10),XCT(38),XQD(29,14),VE(27,120,2),
     3VXX(29,14),VZ(27,42,8,2),ZT(8),ZTT(8),ZTX(8),SAVRGV(10)
      REAL LESION
    2 FORMAT(10F7.3)
    3 FORMAT(F7.4,3I7)
    4 FORMAT(11F7.2)
    5 FORMAT(10I7)
    6 FORMAT(F7.2,I7,2F7.2)
    7 FORMAT(10E7.2)
    8 FORMAT(I7,3E7.2)
    9 FORMAT(F7.2,2I7,F7.2)
  300 READ(5,4,END=200) (FTIME(L),L=1,38)
      READ(5,5)IPRT
      READ(5,3)RIM,LIM,IFIL,IGX
      READ(5,9)RMAX,LIMAX,MAXPRT,LESION
C *** SET VALUES FOR MTEST,N,N1,N3,N4, AND DR
      MTEST=0
      N1=4
      N=N1+9
      N3=N+1
      N4=N1+1
      READ(5,8)IPROF,POW,CUT
      DR=LESION/LIM
      IF(IPROF.EQ.0)DR=RIM/(LIM-.5)
      READ(5,7)DPULSE
      READ(5,5)NTEST,(NRUN(L),L=1,NTEST)
      READ(5, 7)(REPET(L),L=1,NTEST)
      READ(5, 5)(NPULSE(L),L=1,NTEST)
      READ(5,5)ID1,ID2,JD1,JD2,ITYPE
      LPX=1
      IF(NTEST.EQ.1.AND.NPULSE(1).EQ.1)LPX=0
      XDPULS=DPULSE
      XXQ=1.
      IF(DPULSE.GT..3E-8)GO TO 10
C *** ADJUST POWER AND PULSE WIDTH FOR EXPOSURES WITH PULSES LESS THAN
C *** .3E-8 SEC
      XXQ=.3E-8/DPULSE
      POW=POW*DPULSE/.3E-8
      DPULSE=.3E-8
   10 READ(5,4)TO,EDT1,EDT2
```

```
      READ(5,4)TOM,APE,AVL,ACH,ASC,ATS,RCO,RRT,RSC,RPF,WAVEL
      READ(5,4)TAV,TPE,TVL,TCH,TSC,RVL
      AAV=-ALOG(TOM)/TAV
      READ(5, 4) (CONX(L),L=1,6)
      READ(5,4) (VSHX(L),L=1,6)
      READ(5,5) (NPT(L),L=1,38)
      READ(5,2) (XCT(L),L=1,38)
      READ(5,5) (KTT(L),L=1,38)
C *** COMPUTE DT,KM,KT,NP,PTIME,TIME, AND XC
      L1=ALOG(DPULSE)/.69315+29.
      IF(L1.LT.1)L1=1
      IF(L1.GT.38)L1=38
      IF(LPX.EQ.1)GO TO 11
C *** ---SINGLE PULSED EXPOSURES
      XC=XCT(L1)
      NP=NPT(L1)
      KT=KTT(L1)
      DT=DPULSE*(XC-1.)/(XC**NP-1.)
      TIME=DT*(XC**KT-1.)/(XC-1.)
      GO TO 13
C *** ---MULTIPLE PULSED EXPOSURES
   11 XC=1.4
      NP=5
      X1=0.
      DO 12 L=1,NTEST
      IF(X1.LT.NPULSE(L)/REPET(L))X1=NPULSE(L)/REPET(L)
   12 CONTINUE
      TIME=FTIME(L1)*X1
      DT=DPULSE*(XC-1.)/(XC**NP-1.)
      KT=ALOG(1.+TIME*(XC-1.)/DT)/ALOG(XC)+1.
      PTIME=DPULSE/NP
   13 KT=KT+1
      KM=NP+1
      IF(KT.GT.119)WRITE(6,14)KT
   14 FORMAT(1H0,3HKT=,I3,2X,22HTIME DIMENSION TOO LOW)
      IF(KT.GT.119)STOP
C *** CALC. DZ AND I INDICES
      M1=6
      M=2*M1+16
      M2=M/2
      M3=M+1
      IPE=M2-M1+2
      DZ=TPE/M1-1.E-25
      IPA=2
C *** STORE AXIAL DISTANCES TO INTERFACES OF EYE
      ZD(1)=1.E-25
      ZD(2)=TAV
      ZD(3)=ZD(2)+RPE*TPE
      ZD(4)=ZD(3)+(1.-RPE)*TPE
      ZD(5)=ZD(4)+TVL
      ZD(6)=ZD(5)+TCH
      ZD(7)=ZD(6)+TSC
      ZD(8)=ZD(7)+10.
      CALL GRID
      NVL=LPV-IPV+1
C *** CALCULATE AND STORE I,J INDICES AT WHICH TEMPERATURES ARE PRINTED
      ID1=ID1+IPE
```

```
      ID2=ID2+IPE
      IF(ID1.LT.IPA)ID1=IPA
      IF(ID2.GT.M)ID2=M
      IF(JD2.GT.N)JD2=N
      IF(IPRT(1).EQ.0)GO TO 23
      WRITE(6,15)ID1,ID2,JD1,JD2
   15 FORMAT(1H0,5X,4HID1=,I3,3X,4HID2=,I3,3X,4HJD1=,I2,3X,4HJD2=,I2)
      WRITE(6,16)DR,DZ
   16 FORMAT(1H0,5X,3HDR=,E11.4,2X,3HDZ=,E11.4)
      WRITE(6,17)IPA,IPC,IPE,IPS,IPT,IPV,LPA,LPC,LPE,LPS,LPV
   17 FORMAT(1H0,5X,4HIPA=,I3,2X,4HIPC=,I3,2X,4HIPE=,I3,2X,4HIPS=,I3,2X,
     14HIPT=,I3,2X,4HIPV=,I3/1H ,5X,4HLPA=,I3,2X,4HLPC=,I3,2X,4HLPE=,I3,
     22X,4HLPS=,I3,2X,4HLPV=,I3)
      WRITE(6,22)M,M1,N,N1
   22 FORMAT(1H0,5X,2HM=,I2,2X,3HM1=,I2,2X,2HN=,I2,2X,3HN1=,I2)
      WRITE(6,18) (R(J),J=1,N3)
   18 FORMAT(1H0,5X,2HR=/(1H ,5X,10F8.4))
      WRITE(6,19) (Z(I),I=1,M3)
   19 FORMAT(1H0,5X,2HZ=/(1H ,5X,10F8.4))
   23 DO 20 L1=1,NVL
   20 IBLOOD(L1)=IPV+L1-1
C *** CALC. NORMALIZED LASER PROFILES---
      DO 21 L=1,N3
   21 HR(L)=0.
      POX=POW
      CALL IMAGE
      DO 27 J=1,N3
      DO 27 I=1,M3
      V(I,J)=1.E-10
   27 S(I,J)=0.
      READ(5,2)SHB,XFLOW,CFLOW
C *** SET BLOOD FLOW RATES ENTERING AND LEAVING VASCULAR LAYER AS
C *** FUNCTION OF RADIAL DISTANCE
      X2=CFLOW/(3.1416*RVL*RVL)
      DFLOW(1)=0.
      X4=0.
      DO 30 L1=2,6
      X4=X4+.1
   30 DFLOW(L1)=X4
      DO 31 L1=1,6
      XFLOWI(L1)=X2
   31 XFLOWO(L1)=X2
      DO 34 I=1,M3
      DO 34 J=1,N3
   34 VC(I,J,1)=1.E-10
      XPOW=XXQ*POW
      READ(5,8)KTYPEO
      READ(5,8)KTYPE
      L1=KTYPE
      IF(KTYPE.EQ.0)L1=1
      READ(5,7)(TIMEX(K),K=1,L1)
      READ(5,5)II1,II2,II3,JJ1,JJ2
C *** START OF TEMPERATURE CALCULATIONS FOR ONE PULSE. TO BE USED EITHER
C *** FOR MULTIPLE OR SINGLE PULSED EXPOSURES
C     ------------------------------------------------------------
      XT(1)=0.
      DTX=DT
```

```
      KTX=KT+1
      DO 36 K=2,KTX
      XT(K)=XT(K-1)+DT
   36 DT=XC*DT
      IKX=TIME**EDT1+EDT2
      IF(IKX.LT.1)IKX=1
      XX=2*IKX
      K=2
      IHT=2
      ITYPEX=ITYPE
      CALL BLOOD
   38 DT=XT(K)-XT(K-1)
      IF(K.GT.KM)QP=0.
      CALL HTXDEP
      IF(K.GT.2)GO TO 41
      IF(IPRT(2).EQ.0)GO TO 335
      WRITE(6,301)
  301 FORMAT(1H0,13HLASER PROFILE)
      IF(IPROF.EQ.0)WRITE(6,302)RIM
  302 FORMAT(1H0,5X,4HRIM=,E10.3)
      IF(IPROF.EQ.1)WRITE(6,303)SIGMA,RIM,CUT
  303 FORMAT(1H0,5X,6HSIGMA=,E10.3,5X,4HRIM=,E10.3,5X,4HCUT=,E10.3)
      IF(IFIL.EQ.1)WRITE(6,304)RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC,
     1WAVEL
  304 FORMAT(1H0,5X,5HRINT=,E10.3,3X,3HZO=,E10.3,3X,4HFLO=,F6.3/1H ,5X,
     16HCABER=,E10.3,3X,7HCABER2=,F7.0,3X,3HPP=,F6.3/1H ,5X,3HPC=,F6.3,
     23X,3HNB=,F7.3,3X,3HNC=,F7.3/1H ,5X,3HFC=,F6.3,3X,6HWAVEL=,F7.1)
      IF(IFIL.EQ.1)GO TO 306
      IF(IPROF.EQ.2)WRITE(6,305)RINT
  305 FORMAT(1H0,5X,5HRINT=,E10.3)
  306 WRITE(6,307)QP
  307 FORMAT(1H0,5X,3HQP=,E10.3)
      WRITE(6,308) (HR(J),J=1,N)
  308 FORMAT(1H0,5X,3HHR=/(1H ,10X,10E10.3))
  335 IF(IPRT(3).EQ.0)GO TO 340
      WRITE(6,309)
  309 FORMAT(1H0,19HDATA IDENTIFICATION)
      WRITE(6,310) (REPET(L),L=1,NTEST)
  310 FORMAT(1H0,5X,6HREPET=/(1H ,5X,10E10.3))
      WRITE(6,311) (NPULSE(L),L=1,NTEST)
  311 FORMAT(1H0,5X,7HNPULSE=/(1H ,5X,10I8))
      WRITE(6,312)AAV,ACH,APE,ASC,ATS,RCO,RRT,RPE,TOM,AVL,TAV,TCH,TPE,
     1TSC,TVL,IGX,IFIL,IPROF,LIM,NTEST,POW,XDPULS,RIM,RMAX,TIME,CFLOW,
     2XFLOW,SHB,EDT1,EDT2,DT,KM,KT,PTIME,XC,IKX,AP,APE1,APE2,IG,RVL,
     3PUPIL,TO,LIMAX,MAXPRT
  312 FORMAT(1H0,5X, 4HAAV=,F7.1,2X,4HACH=,F7.0,2X,4HAPE=,F7.0,2X,4HASC=
     1,F7.0,2X,4HATS=,F7.0/1H ,5X,4HRCO=,F7.4,2X,4HRRT=,F7.4,2X,4HRPE=,
     2F7.4,2X,4HTOM=,F7.4,2X,4HAVL=,F7.0/1H ,5X,4HTAV=,E9.3,2X,4HTCH=,
     3E9.3,2X,4HTPE=,E9.3,2X,4HTSC=,E9.3,2X,4HTVL=,E9.3/1H ,5X,4HIGX=,I2
     4,2X,5HIFIL=,I2,2X,6HIPROF=,I2,2X,4HLIM=,I2,2X,6HNTEST=,I2/1H ,5X,
     54HPOW=,E9.3,2X,7HDPULSE=,E9.3,2X,4HRIM=,F7.4,2X,5HRMAX=,F7.4,2X,
     65HTIME=,E9.3/1H ,5X,6HCFLOW=,F7.4,2X,6HXFLOW=,F7.4,2X,4HSHB=,F7.2,
     72X,5HEDT1=,F7.4,2X,5HEDT2=,F7.4/1H ,5X,3HDT=,E9.3,2X,3HKM=,I3,2X,
     83HKT=,I3,2X,6HPTIME=,E9.3,2X,3HXC=,F5.1/1H ,5X,4HIKX=,I2,2X,3HAP=,
     9F7.4,2X,5HAPE1=,F8.2,2X,5HAPE2=,F8.2,2X,3HIG=,I3/1H ,5X,4HRVL=,
     1F6.3,2X,6HPUPIL=,F6.3,2X,3HTO=,F5.1,2X,6HLIMAX=,I2,2X,7HMAXPRT=,
     2I2)
```

```
340 IF(IPRT(4).EQ.0)GO TO 355
    WRITE(6,313)
313 FORMAT(1H0,30HBLOOD FLOW AND HEAT DEPOSITION)
    WRITE(6,314) (FLOWI(J),J=1,JVL)
314 FORMAT(1H0,5X,6HFLOWI=/(1H ,5X,10E10.3))
    WRITE(6,315) (FLOWX(J),J=1,JVL)
315 FORMAT(1H0,5X,6HFLOWX=/(1H ,5X,10E10.3))
    WRITE(6,316)
316 FORMAT(1H )
    DO 318 I=IPA,M
    WRITE(6,317) (S(I,J),J=1,N)
317 FORMAT(1H ,5X,2HS=,10F8.3)
318 CONTINUE
355 IF(IPRT(5).EQ.0)GO TO 41
    WRITE(6,319)
319 FORMAT(1H0,17HTEMPERATURE RISES)
    JCNT=JD2-JD1+1
    IF(JCNT.GT.9)GO TO 40
    GO TO 41
 40 JJCNT=JCNT-9
    JJD2=JD2-JJCNT
    JJD2P1=JJD2+1
 41 IF(IPRT(5).EQ.0)GO TO 356
    WRITE(6,42)XT(K),K
 42 FORMAT(1H0,5X,5HTIME=,E11.4,3X,2HK=,I3)
C *** CALCULATE TEMPERATURE RISE(MATRIX REDUCTION ALGORITHM)
C *** COLUMNS(NORMAL)-------------------
356 IK=1
 43 DO 45 I=IPA,M
    W=XX*VSH(I)/DT
    DO 44 J=1,N
    FXC(J)=W+CON(I)*B(J,2)-BV(J,2)*IV(I)-BB*IAB(I,J)
    IF(J.GT.1)FXC(J)=FXC(J)+(CON(I)*B(J,1)+BV(J,1)*IV(I))*CXC(J-1)
    CXC(J)=-(CON(I)*B(J,3)+BV(J,3)*IV(I))/FXC(J)
    SUM=(W-(A(I,2)-BV(J,2)*IV(I)-BB*IAB(I,J)))*V(I,J)+A(I,1)*V(I-1,J)+
   1A(I,3)*V(I+1,J)+S(I,J)
    DXC(J)=SUM/FXC(J)
    IF(J.GT.1)DXC(J)=(SUM+(CON(I)*B(J,1)+BV(J,1)*IV(I))*DXC(J-1))/FXC(
   1J)
 44 CONTINUE
    VX=0.
    DO 45 L=1,N
    J=N+1-L
    VX=DXC(J)-CXC(J)*VX
 45 VXX(I,J)=VX
    DO 46 I=IPA,M
    DO 46 J=1,N
 46 V(I,J)=VXX(I,J)
C *** ROWS(NORMAL)---------------------
    CXR(IPA-1)=0.
    DO 50 J=1,N
    DO 48 I=IPA,M
    W=XX*VSH(I)/DT
    FXR(I)=W+A(I,2)-BV(J,2)*IV(I)-BB*IAB(I,J)+A(I,1)*CXR(I-1)
    CXR(I)=-A(I,3)/FXR(I)
    SUM=(W-(CON(I)*B(J,2)-BV(J,2)*IV(I)-BB*IAB(I,J)))*V(I,J)+(CON(I)*
   1B(J,3)+BV(J,3)*IV(I))*V(I,J+1)+S(I,J)
```

```
      IF(J.GT.1)SUM=SUM+(CON(I)*B(J,1)+BV(J,1)*IV(I))*V(I,J-1)          RE100286
      DXR(I)=SUM/FXR(I)                                                  RE100287
      IF(I.GT.IPA)DXR(I)=(SUM+A(I,1)*DXR(I-1))/FXR(I)                    RE100288
   48 CONTINUE                                                           RE100289
      VX=0.                                                              RE100290
      DO 50 L=IPA,M                                                      RE100291
      I=M+IPA-L                                                          RE100292
      VX=DXR(I)-CXR(I)*VX                                                RE100293
      VC(I,J,K)=VX                                                       RE100294
   50 VXX(I,J)=VX                                                        RE100295
      DO 51 I=IPA,M                                                      RE100296
      DO 51 J=1,N                                                        RE100297
   51 V(I,J)=VXX(I,J)                                                    RE100298
      IK=IK+1                                                            RE100299
C *** RECYCLE TEMPERATURE CALCULATIONS                                   RE100300
      IF(IK.LE.IKX)GO TO 43                                              RE100301
      IF(K.EQ.KM)GO TO 62                                                RE100302
      IF(ITYPEX.LT.ITYPE.AND.K.LT.KT)GO TO 66                            RE100303
   62 IF(IPRT(5).EQ.0)GO TO 357                                          RE100304
      WRITE(6,63) (R(J),J=JD1,JD2)                                       RE100305
   63 FORMAT(1H ,13X,2HR=,9F13.5/1H ,15X,30H----------------------------RE100306
     1--)                                                                RE100307
      DO 65 I=ID1,ID2                                                    RE100308
      X1=Z(I)-Z(IPE)+DZ/2.                                               RE100309
      IF(JCNT.GT.9)GO TO 57                                              RE100310
      WRITE(6,64)X1,(VC(I,J,K),J=JD1,JD2)                                RE100311
      GO TO 65                                                           RE100312
   57 WRITE(6,64)X1,(VC(I,J,K),J=JD1,JJD2)                               RE100313
      WRITE(6,64)X1,(VC(I,J,K),J=JJD2P1,JD2)                             RE100314
   64 FORMAT(1H ,3X,2HZ=,F8.5,2X,1P9E13.6)                               RE100315
   65 CONTINUE                                                           RE100316
  357 ITYPEX=0                                                           RE100317
   66 K=K+1                                                              RE100318
      ITYPEX=ITYPEX+1                                                    RE100319
      IF(K.LE.KT)GO TO 38                                                RE100320
      ITYPEX=ITYPE                                                       RE100321
      IF(IPRT(6).EQ.0)GO TO 365                                          RE100322
      WRITE(6,320)                                                       RE100323
  320 FORMAT(1H0,28HNORMALIZED TEMPERATURE RISES)                        RE100324
      DO 70 K=2,KT                                                       RE100325
      IF(K.EQ.KM)GO TO 67                                                RE100326
      IF(ITYPEX.LT.ITYPE.AND.K.LT.KT)GO TO 70                            RE100327
   67 X1=1.                                                              RE100328
      WRITE(6,321)XT(K),K,X1                                             RE100329
  321 FORMAT(1H0,5X,5HTIME=,E11.4,3X,2HK=,I3,3X,6HPOWER=,E11.4,5HWATTS)  RE100330
      WRITE(6,63) (R(J),J=JD1,JD2)                                       RE100331
      JCNT=JD2-JD1+1                                                     RE100332
      IF(JCNT.GT.9)GO TO 380                                             RE100333
      GO TO 381                                                          RE100334
  380 JJCNT=JCNT-9                                                       RE100335
      JJD2=JD2-JJCNT                                                     RE100336
      JJD2P1=JJD2+1                                                      RE100337
  381 DO 69 I=ID1,ID2                                                    RE100338
      DO 68 J=JD1,JD2                                                    RE100339
   68 V(I,J)=VC(I,J,K)/POW                                               RE100340
      X1=Z(I)-Z(IPE)+DZ/2.                                               RE100341
      IF(JCNT.GT.9)GO TO 382                                             RE100342
```

```
      WRITE(6,64)X1,(V(I,J),J=JD1,JD2)
      GO TO 69
  382 WRITE(6,64)X1,(V(I,J),J=JD1,JJD2)
      WRITE(6,64)X1,(V(I,J),J=JJD2P1,JD2)
   69 CONTINUE
      ITYPEX=0
   70 ITYPEX=ITYPEX+1
C *** READ NORMALIZED TEMPERATURE RISES TS OF GRANULES FOR .3E-8 PULSE
C *** CALCULATE NORMALIZED RISES XPD FOR ACTUAL PULSE
  330 FORMAT(1H0,61HDIMENSION OF ARRAYS ASSOCIATED WITH ARGUMENT LIJ IS
     1TOO SMALL)
  365 READ(5,8)LTMAX
      DO 71 L1=1,LTMAX
   71 TS(L1)=1.
      READ(5,2)(TS(L),L=1,LTMAX,10)
      CALL MXGRAN
      DO 72 L=1,KT
   72 XPD(L)=AP*XPD(L)+1.-AP
      READ(5,4)(DAMAGE(L2,1),DAMAGE(L2,2),L2=1,2),TSTEAM,DTSTM
      WRITE(6,73)WAVEL,TSTEAM,DAMAGE(1,1),DAMAGE(1,2),DAMAGE(2,1),
     1DAMAGE(2,2)
   73 FORMAT(1H0,5X,11HWAVELENGTH=,F7.1,2HNM,3X,7HTSTEAM=,F6.0,3X,7HDAMA
     1GE=,4F9.0)
C *** CALCULATE I,J INDICES AT WHICH DAMAGE CALCULATIONS ARE TO BE MADE
      JM=0
      DO 74 J=1,N
      IF(R(J).LT.RMAX+.000001)JM=J+1
   74 CONTINUE
      X1=0.
      DO 75 I=IPA,M
      IF(VC(I,1,KM).GT.X1)IMAX=I
      IF(VC(I,1,KM).GT.X1)X1=VC(I,1,KM)
   75 CONTINUE
      L=0
      GO TO (366,367,368),MAXPRT
  366 LIMAX1=2*LIMAX
      LIMAX2=0
      GO TO 369
  367 LIMAX1=LIMAX
      LIMAX2=LIMAX
      GO TO 369
  368 LIMAX1=0
      LIMAX2=2*LIMAX
  369 ID1=IMAX-LIMAX1
      ID2=IMAX+LIMAX2
      IF(ID2.GT.28)ID2=28
      DO 76 I=ID1,ID2
      DO 76 J=1,JM
      L=L+1
      ID(L)=I
   76 JD(L)=J
      LIJ=(ID2-ID1+1)*JM
      DO 385 LL15=1,10
  385 SAVRGV(LL15)=0.
      IF(LPX.EQ.0)GO TO 125
      IF(LIJ.GT.27)WRITE(6,330)
      IF(LIJ.GT.27)GO TO 300
```

```
      IF(IPRT(8).EQ.0)GO TO 370
C *** TEMPERATURE AND DAMAGE EVALUATIONS FOR MULTIPLE PULSES
C     -----------------------------------------------------------
C *** EVALUATE  TEMPERATURE RISES WITH AND WITHOUT GRANULES
      DO 77 L=1,LIJ
      I=ID(L)
      J=JD(L)
      VE(L,1,1)=0.
      VE(L,1,2)=0.
      DO 77 K=2,KT
      VE(L,K,1)=VC(I,J,K)
      VE(L,K,2)=VC(I,J,K)
      IF(I.NE.IG)GO TO 77
      VE(L,K,2)=XPD(K)*VC(I,J,K)
      IF(VE(L,K,1).LT..0)VE(L,K,1)=0.
      IF(VE(L,K,2).LT..0)VE(L,K,2)=0.
   77 CONTINUE
      X60=(XC-1.)/DEX
      X61=ALOG(XC)
      XSTEAM=TSTEAM
  370 L13=0
  371 L13=L13+1
      X3=DPULSE+(NPULSE(L13)-1)/REPET(L13)
      WRITE(6,78)NRUN(L13),X3,XDPULS,NPULSE(L13),REPET(L13)
   78 FORMAT(1H0,5X,5HNRUN=,I3,2X,13HTRAIN LENGTH=,E10.3,3HSEC,2X,12HPUL
     1SE WIDTH=,E10.3,3HSEC/1H ,5X,17HNUMBER OF PULSES=,I5,3X,16HREPETIT
     2ION RATE=,E10.3,10HPULSES/SEC)
      IF(IFIL.EQ.0)GO TO 80
      WRITE(6,79)RIM,LESION
   79 FORMAT(1H ,5X,12HBEAM RADIUS=,E10.3,2HCM,5X,14HLESION RADIUS=,E10.
     13,2HCM)
      GO TO 82
   80 WRITE(6,81)RIM,LESION
   81 FORMAT(1H ,5X,13HIMAGE RADIUS=,E10.3,2HCM,5X,14HLESION RADIUS=,E10
     1.3,2HCM)
   82 IF(IPRT(8).EQ.0)GO TO 108
      TC=1./REPET(L13)
      NPL=NPULSE(L13)
      KX=NP+3
      IN=1
   83 IF(NPL/IN.LT.20)GO TO 84
      IN=IN+2
      GO TO 83
   84 X1=NPL
      INX=.5+X1/IN
      L1=ALOG(DPULSE)/.69315+29.
      IF(L1.LT.1)L1=1
      INXX=PTIME(L1)*INX
C *** STORE TIME INTERVALS AND LOGS OF INTERVALS FOR DAMAGE CALCULATIONS
      ZTX(1)=PTIME
      ZT(1)=PTIME/2.
      ZTT(1)=ALOG(IN*PTIME)
      DO 85 L3=2,NP
      ZTT(L3)=ALOG(IN*PTIME)
      ZTX(L3)=ZTX(L3-1)+PTIME
   85 ZT(L3)=ZT(L3-1)+PTIME
      L1=NP+1
```

```
      X3=(TC-DPULSE)/(KX-NP)
      ZTX(L1)=DPULSE+X3
      ZT(L1)=DPULSE+X3/2.
      ZTT(L1)=ALOG(IN*X3)
      L1=L1+1
      DO 86 L3=L1,KX
      ZTT(L3)=ALOG(IN*X3)
      ZTX(L3)=ZTX(L3-1)+X3
   86 ZT(L3)=ZT(L3-1)+X3
C *** CALCULATE TEMPERATURE RISES ASSOCIATED WITH L3-TH TIME INTERVAL
C *** FOLLOWING (L6-.5)*IN-.5 PULSE
      DO 95 L=1,LIJ
      DO 95 L3=1,KX
      X1=0.
      X2=0.
      L1=1+IN/2
      L7=1
   87 X3=(L7-1)*TC+ZT(L3)
      K=ALOG(X3*X60+1.)/X61+1.
      X5=VE(L,K,1)+(X3-XT(K))*(VE(L,K+1,1)-VE(L,K,1))/(XT(K+1)-XT(K))
      X1=X1+X5
      X3=(L7-1)*TC+ZTX(L3)
      K=ALOG(X3*X60+1.)/X61+1.
      X2=X2+VE(L,K,2)+(X3-XT(K))*(VE(L,K+1,2)-VE(L,K,2))/(XT(K+1)-XT(K))
      IF(X5.LT..0001*X1)GO TO 88
      L7=L7+1
      IF(L7.LE.L1)GO TO 87
   88 VZ(L,1,L3,1)=X1
      VZ(L,1,L3,2)=X2
      DO 93 L6=2,INXX
      IF(X5.LT..0001*X1)GO TO 91
      X1=VZ(L,L6-1,L3,1)
      X2=VZ(L,L6-1,L3,2)
      L2=L1+1
      L1=L1+IN
      L7=L2
   90 X3=(L7-1)*TC+ZT(L3)
      K=ALOG(X3*X60+1.)/X61+1.
      X5=VE(L,K,1)+(X3-XT(K))*(VE(L,K+1,1)-VE(L,K,1))/(XT(K+1)-XT(K))
      X1=X1+X5
      X3=(L7-1)*TC+ZTX(L3)
      K=ALOG(X3*X60+1.)/X61+1.
      X2=X2+VE(L,K,2)+(X3-XT(K))*(VE(L,K+1,2)-VE(L,K,2))/(XT(K+1)-XT(K))
      IF(X5.LT..0001*X1)GO TO 91
      L7=L7+1
      IF(L7.LE.L1)GO TO 90
   91 VZ(L,L6,L3,1)=X1
   93 VZ(L,L6,L3,2)=X2
      L1=INX+1
      DO 94 L6=L1,INXX
      L8=L6-INX
      VZ(L,L6,L3,1)=VZ(L,L6,L3,1)-VZ(L,L8,L3,1)
   94 VZ(L,L6,L3,2)=VZ(L,L6,L3,2)-VZ(L,L8,L3,2)
   95 CONTINUE
C *** DAMAGE CALCULATIONS --------------
      WRITE(6,375)
  375 FORMAT(1H0,31HPREDICTED THRESHOLD LASER POWER)
```

```
      TSTEAM=XSTEAM                                                     RE100514
      XQ=0.                                                             RE100515
   96 WRITE(6,130)TSTEAM                                                RE100516
      DO 104 L=1,LIJ                                                    RE100517
      I=ID(L)                                                           RE100518
      J=JD(L)                                                           RE100519
      IF(VZ(L,INX,NP,1).LT..001)QD(I,J)=1.E+20                          RE100520
      IF(VZ(L,INX,NP,1).LT..001)GO TO 104                               RE100521
      L9=10.*(.4+EXP(-.0014*DPULSE))/VZ(L,INX,NP,1)                     RE100522
      CQ=L9+1.                                                          RE100523
      X10=70.*(.4+EXP(-.0014*DPULSE))/VZ(L,INX,NP,1)                    RE100524
      IF(L9.EQ.0)CQ=X10                                                 RE100525
      LLT=0                                                             RE100526
      LGT=0                                                             RE100527
   99 DAMC=0.                                                           RE100528
      L6=1                                                              RE100529
  100 DO 101 L3=1,KX                                                    RE100530
      X3=0.                                                             RE100531
      IF(VZ(L,L6,L3,2)*CQ.GT.TSTEAM-T0)X3=1.E+30                        RE100532
      IF(VZ(L,L6,L3,2)*CQ.GT.TSTEAM-T0)GO TO 101                        RE100533
      X50=VZ(L,L6,L3,1)*CQ+273.+T0                                      RE100534
      IF(X50.LT.317.)GO TO 101                                          RE100535
      X1=ZTT(L3)+DAMAGE(1,1)-DAMAGE(1,2)/X50                            RE100536
      IF(X50.GT.323.)X1=ZTT(L3)+DAMAGE(2,1)-DAMAGE(2,2)/X50             RE100537
      IF(X1.GT.0.)X3=1.01                                               RE100538
      IF(X1.GT.0.)GO TO 101                                             RE100539
      X3=EXP(X1)                                                        RE100540
  101 DAMC=DAMC+X3                                                      RE100541
      IF(DAMC.GT.1.)GO TO 102                                           RE100542
C *** INCREASE TIME INDICES AND CONTINUE                                RE100543
      L6=L6+1                                                           RE100544
      IF(L6.LE.INXX)GO TO 100                                           RE100545
C *** ADJUST LASER POWER TO YIELD THRESHOLD DAMAGE AT GIVEN POINT       RE100546
      IF(LGT.EQ.1)CQ=1.02*CQ                                            RE100547
      IF(LGT.EQ.1)GO TO 103                                             RE100548
      LLT=1                                                             RE100549
      CQ=1.04*CQ                                                        RE100550
      GO TO 99                                                          RE100551
  102 IF(LLT.EQ.1)CQ=.98*CQ                                             RE100552
      IF(LLT.EQ.1)GO TO 103                                             RE100553
      LGT=1                                                             RE100554
      CQ=.96*CQ                                                         RE100555
      GO TO 99                                                          RE100556
  103 QD(I,J)=CQ*POX                                                    RE100557
  104 CONTINUE                                                          RE100558
      WRITE(6,63)(R(J),J=1,JM)                                          RE100559
      DO 97 I=ID1,ID2                                                   RE100560
      DO 97 J=1,JM                                                      RE100561
   97 XQD(I,J)=QD(I,J)*XXQ                                              RE100562
      DO 106 I=ID1,ID2                                                  RE100563
      X1=Z(I)-Z(IPE)+DZ/2.                                              RE100564
      IF(JM.GT.9)GO TO 98                                               RE100565
      WRITE(6,105)X1,(XQD(I,J),J=1,JM)                                  RE100566
      GO TO 106                                                         RE100567
   98 WRITE(6,105)X1,(XQD(I,J),J=1,9)                                   RE100568
      WRITE(6,105)X1,(XQD(I,J),J=10,JM)                                 RE100569
  105 FORMAT(1H ,2X,2HZ=,F7.5,1X,3HQD=,1P9E13.6)                        RE100570
```

```
  106 CONTINUE
      X2=(XQ-QD(IMAX,1))/QD(IMAX,1)
      X3=X2*X2
      IF(X3.LT..0001)GO TO 108
      TSTEAM=TSTEAM+DTSTM
      XQ=QD(IMAX,1)
      GO TO 96
  108 IF(KTYPE.EQ.0)GO TO 174
C *** CALCULATE AND STORE (MULTIPLE PULSE EXPOSURE) TEMPERATURES FOR
C *** PLOTTING PROFILES
      TC=1./REPET(L13)
      NPL=NPULSE(L13)
      WRITE(6,139)
      DO 123 L15=1,KTYPE
      IF(TIMEX(L15).GT.XT(KT))GO TO 123
      RGV=0.
      L2=TIMEX(L15)/TC
      DTIME=TIMEX(L15)-L2*TC
      L2=L2+1
      DO 116 I=II1,II2
      DO 116 J=JJ1,JJ2
      X1=0.
      DO 113 L6=1,L2
      K=ALOG((DTIME+(L6-1)*TC)*X60+1.)/X61+1.
      X2=(DTIME+(L6-1)*TC-XT(K))/(XT(K+1)-XT(K))
  113 X1=X1+VC(I,J,K)+X2*(VC(I,J,K+1)-VC(I,J,K))
      V(I,J)=X1
      L3=L2-NPL
      IF(L3.LE.0)GO TO 115
      X1=0.
      DO 114 L6=1,L3
      K=ALOG((DTIME+(L6-1)*TC)*X60+1.)/X61+1.
      X2=(DTIME+(L6-1)*TC-XT(K))/(XT(K+1)-XT(K))
  114 X1=X1+VC(I,J,K)+X2*(VC(I,J,K+1)-VC(I,J,K))
      V(I,J)=V(I,J)-X1
  115 IF(V(I,J).GT.RGV)RGV=V(I,J)
  116 CONTINUE
      SAVRGV(L15)=RGV
      IF(KTYPEO.EQ.1)GO TO 121
      WRITE(7,117)NRUN(L13),NPULSE(L13),REPET(L13)
  117 FORMAT(2I7,E10.4)
      WRITE(7,118)XDPULS,WAVEL,RIM
  118 FORMAT(7E11.4)
      WRITE(7,119)II1,II2,II3,JJ1,JJ2
  119 FORMAT(5I7)
      WRITE(7,119)N3,M3
      WRITE(7,120)(R(J),J=1,N3)
  120 FORMAT(10F8.4)
      WRITE(7,120)(Z(I),I=1,M3)
      WRITE(7,118)TIMEX(L15)
  121 WRITE(6,141)TIMEX(L15)
      WRITE(6,63)(R(J),J=JJ1,JJ2)
      JCNT=JJ2-JJ1+1
      IF(JCNT.GT.9)GO TO 390
      GO TO 391
  390 JJCNT=JCNT-9
      JJJ2=JJ2-JJCNT
```

```
      JJJ2P1=JJJ2+1                                                     RE100628
  391 DO 122 I=II1,II2                                                  RE100629
      X1=Z(I)-Z(IPE)+DZ/2.                                              RE100630
      IF(JCNT.GT.9)GO TO 392                                            RE100631
      WRITE(6,64)X1,(V(I,J),J=JJ1,JJ2)                                  RE100632
      GO TO 393                                                         RE100633
  392 WRITE(6,64)X1,(V(I,J),J=JJ1,JJJ2)                                 RE100634
      WRITE(6,64)X1,(V(I,J),J=JJJ2P1,JJ2)                               RE100635
  393 IF(KTYPEO.EQ.1)GO TO 122                                          RE100636
      WRITE(7,137) (V(I,J),J=JJ1,JJ2)                                   RE100637
  122 CONTINUE                                                          RE100638
  123 CONTINUE                                                          RE100639
      RGV=0.                                                            RE100640
      DO 395 LL15=1,KTYPE                                               RE100641
      IF(SAVRGV(LL15).GT.RGV)RGV=SAVRGV(LL15)                           RE100642
  395 CONTINUE                                                          RE100643
      WRITE(7,396)                                                      RE100644
  396 FORMAT(22HMAX RGV CARD(S) FOLLOW)                                 RE100645
      DO 397 LL15=1,KTYPE                                               RE100646
  397 WRITE(7,137)RGV                                                   RE100647
      GO TO 174                                                         RE100648
  124 FORMAT(1H ,5X,1P9E13.6)                                           RE100649
  137 FORMAT(6E13.6)                                                    RE100650
  139 FORMAT(1H0,35HTEMPERATURE RISES AT SELECTED TIMES)                RE100651
  141 FORMAT(1H0,5X,5HTIME=,E11.4)                                      RE100652
  145 IF(L13.EQ.NTEST)GO TO 300                                         RE100653
      GO TO 371                                                         RE100654
C *** DAMAGE CALCULATIONS FOR SINGLE PULSE                                RE100655
C     ---------------------------------------------                      RE100656
  125 WRITE(6,126)NRUN(1),XDPULS,NPULSE(1)                              RE100657
  126 FORMAT(1H0,5X,5HNRUN=,I3,2X,12HPULSE WIDTH=,E10.3,2X,17HNUMBER OF RE100658
     1PULSES=,I5)                                                       RE100659
      IF(IFIL.EQ.0)GO TO 127                                            RE100660
      WRITE(6,79)RIM,LESION                                             RE100661
      GO TO 128                                                         RE100662
  127 WRITE(6,81)RIM,LESION                                             RE100663
  128 IF(IPRT(8).EQ.0)GO TO 150                                         RE100664
      WRITE(6,375)                                                      RE100665
      XQ=0.                                                             RE100666
  129 WRITE(6,130)TSTEAM                                                RE100667
  130 FORMAT(1H0,5X,7HTSTEAM=,F7.0/1H ,5X,10H----------)                RE100668
      DO 138 I=ID1,ID2                                                  RE100669
      DO 138 J=1,JM                                                     RE100670
      IF(VC(I,J,KM).LT..001)QD(I,J)=1.0E+20                             RE100671
      IF(VC(I,J,KM).LT..001)GO TO 138                                   RE100672
      L9=10.*(.4+EXP(-.0014*DPULSE))/VC(I,J,KM)                         RE100673
      CQ=L9+1.                                                          RE100674
      X10=70.*(.4+EXP(-.0014*DPULSE))/VC(I,J,KM)                        RE100675
      IF(L9.EQ.0)CQ=X10                                                 RE100676
      LLT=0                                                             RE100677
      LGT=0                                                             RE100678
  131 DAMC=0.                                                           RE100679
      K=2                                                               RE100680
  132 X13=ALOG(XT(K)-XT(K-1))                                           RE100681
      VPX=(VC(I,J,K)+VC(I,J,K-1))/2.                                    RE100682
      X3=0.                                                             RE100683
      IF(I.NE.IG)GO TO 133                                              RE100684
```

```
      IF(VPX*XPD(K)*CQ.GT.TSTEAM-TO) X3=1.E+30                          RE100685
      IF(VPX*XPD(K)*CQ.GT.TSTEAM-TO)GO TO 134                           RE100686
  133 X50=VPX*CQ+273.+TO                                                RE100687
      IF(X50.LT.317.)GO TO 134                                          RE100688
      X1=X13+DAMAGE(1,1)-DAMAGE(1,2)/X50                                RE100689
      IF(X50.GT.323.) X1=X13+DAMAGE(2,1)-DAMAGE(2,2)/X50                RE100690
      IF(X1.GT.0.)X3=1.01                                               RE100691
      IF(X1.GT.0.)GO TO 134                                             RE100692
      X3=EXP(X1)                                                        RE100693
  134 DAMC=DAMC+X3                                                      RE100694
      IF(DAMC.GE.1.)GO TO 135                                           RE100695
      K=K+1                                                             RE100696
      IF(K.LT.KT)GO TO 132                                              RE100697
C *** ADJUST LASER POWER TO YIELD THRESHOLD DAMAGE AT GIVEN POINT       RE100698
      IF(LGT.EQ.1)CQ=1.02*CQ                                            RE100699
      IF(LGT.EQ.1)GO TO 136                                             RE100700
      LLT=1                                                             RE100701
      CQ=1.04*CQ                                                        RE100702
      GO TO 131                                                         RE100703
  135 IF(LLT.EQ.1)CQ=.98*CQ                                             RE100704
      IF(LLT.EQ.1)GO TO 136                                             RE100705
      LGT=1                                                             RE100706
      CQ=.96*CQ                                                         RE100707
      GO TO 131                                                         RE100708
  136 QD(I,J)=CQ*POX                                                    RE100709
  138 CONTINUE                                                          RE100710
      WRITE(6,63) (R(J),J=1,JM)                                         RE100711
      DO 140 I=ID1,ID2                                                  RE100712
      DO 140 J=1,JM                                                     RE100713
  140 XQD(I,J)=QD(I,J)*XXQ                                              RE100714
      DO 143 I=ID1,ID2                                                  RE100715
      X1=Z(I)-Z(IPE)+DZ/2.                                              RE100716
      IF(JM.GT.9)GO TO 142                                              RE100717
      WRITE(6,105)X1,(XQD(I,J),J=1,JM)                                  RE100718
      GO TO 143                                                         RE100719
  142 WRITE(6,105)X1,(XQD(I,J),J=1,9)                                   RE100720
      WRITE(6,105)X1,(XQD(I,J),J=10,JM)                                 RE100721
  143 CONTINUE                                                          RE100722
      X2=(XQ-QD(IMAX,1))/QD(IMAX,1)                                     RE100723
      X3=X2*X2                                                          RE100724
      IF(X3.LT..0001)GO TO 150                                          RE100725
      TSTEAM=TSTEAM+DTSTM                                               RE100726
      XQ=QD(IMAX,1)                                                     RE100727
      GO TO 129                                                         RE100728
  150 IF(KTYPE.EQ.0)GO TO 174                                           RE100729
C *** CALCULATE AND STORE (SINGLE PULSE EXPOSURE) TEMPERATURES FOR      RE100730
C *** PLOTTING PROFILES                                                 RE100731
      WRITE(6,139)                                                      RE100732
      DO 170 L15=1,KTYPE                                                RE100733
      RGV=0.                                                            RE100734
      DTIME=TIMEX(L15)                                                  RE100735
      K=ALOG(DTIME*(XC-1.)/DTX+1.)/ALOG(XC)+1.                          RE100736
      IF(K+1.GT.KT)GO TO 170                                            RE100737
      X1=(DTIME-XT(K))/(XT(K+1)-XT(K))                                  RE100738
      DO 166 I=II1,II2                                                  RE100739
      DO 166 J=JJ1,JJ2                                                  RE100740
      V(I,J)=VC(I,J,K)+X1*(VC(I,J,K+1)-VC(I,J,K))                       RE100741
```

```
      IF(V(I,J).GT.RGV)RGV=V(I,J)                                       RE100742
  166 CONTINUE                                                          RE100743
      SAVRGV(L15)=RGV                                                   RE100744
      IF(KTYPEO.EQ.1)GO TO 167                                          RE100745
      WRITE(7,117)NRUN(1),NPULSE(1),REPET(1)                            RE100746
      WRITE(7,118)XDPULS,WAVEL,RIM                                      RE100747
      WRITE(7,119)II1,II2,II3,JJ1,JJ2                                   RE100748
      WRITE(7,119)N3,M3                                                 RE100749
      WRITE(7,120) (R(J),J=1,N3)                                        RE100750
      WRITE(7,120) (Z(I),I=1,M3)                                        RE100751
      WRITE(7,118)TIMEX(L15)                                            RE100752
  167 WRITE(6,141)TIMEX(L15)                                            RE100753
      WRITE(6,63) (R(J),J=JJ1,JJ2)                                      RE100754
      JCNT=JJ2-JJ1+1                                                    RE100755
      IF(JCNT.GT.9)GO TO 400                                            RE100756
      GO TO 401                                                         RE100757
 400  JJCNT=JCNT-9                                                      RE100758
      JJJ2=JJ2-JJCNT                                                    RE100759
      JJJ2P1=JJJ2+1                                                     RE100760
 401  DO 168 I=II1,II2                                                  RE100761
      X1=Z(I)-Z(IPE)+DZ/2.                                              RE100762
      IF(JCNT.GT.9)GO TO 402                                            RE100763
      WRITE(6,64)X1,(V(I,J),J=JJ1,JJ2)                                  RE100764
      GO TO 403                                                         RE100765
 402  WRITE(6,64)X1,(V(I,J),J=JJ1,JJJ2)                                 RE100766
      WRITE(6,64)X1,(V(I,J),J=JJJ2P1,JJ2)                               RE100767
  403 IF(KTYPEO.EQ.1)GO TO 168                                          RE100768
      WRITE(7,137) (V(I,J),J=JJ1,JJ2)                                   RE100769
  168 CONTINUE                                                          RE100770
  170 CONTINUE                                                          RE100771
      RGV=0.                                                            RE100772
      DO 405 LL15=1,KTYPE                                               RE100773
      IF(SAVRGV(LL15).GT.RGV)RGV=SAVRGV(LL15)                           RE100774
  405 CONTINUE                                                          RE100775
      WRITE(7,396)                                                      RE100776
      DO 406 LL15=1,KTYPE                                               RE100777
  406 WRITE(7,137)RGV                                                   RE100778
C *** INTERPOLATE AXIAL EXTENT OF DAMAGE                                RE100779
  174 I5=0                                                              RE100780
      I6=0                                                              RE100781
      IF(ID1.EQ.ID2)GO TO 182                                           RE100782
      DO 175 I=ID1,ID2                                                  RE100783
      L1=ID1+ID2-I                                                      RE100784
      IF(QD(L1,1).GT.POX)I5=L1                                          RE100785
      IF(QD(L1,1).LT.POX)I6=L1                                          RE100786
      IF(QD(I,1).GT.POX)I7=I                                            RE100787
      IF(QD(I,1).LT.POX)I8=I                                            RE100788
  175 CONTINUE                                                          RE100789
      IF(IPRT(9).EQ.0)GO TO 182                                         RE100790
      WRITE(6,350)                                                      RE100791
  350 FORMAT(1H0,22HAXIAL EXTENT OF DAMAGE)                             RE100792
      IF(I5.EQ.0)WRITE(6,176)                                           RE100793
  176 FORMAT(1H0,5X,45HDEPTHS OF DAMAGE BEYOND BOTH SPECIFIED DEPTHS)   RE100794
      IF(I5.EQ.0)GO TO 182                                              RE100795
      IF(I6.EQ.0)GO TO 190                                              RE100796
      IF(I5.GE.I6)GO TO 178                                             RE100797
      X2=ALOG(QD(I6,1)/QD(I5,1))/(Z(I6)-Z(I5))                          RE100798
```

```
      X1=QD(I5,1)                                                       RE100799
      X3=ALOG(POX/X1)/X2+Z(I5)-Z(IPE)+DZ/2.                             RE100800
      WRITE(6,177)X3                                                    RE100801
  177 FORMAT(1H0,5X,24HMINIMUM DEPTH OF DAMAGE=,E10.3,2HCM)             RE100802
  178 IF(I8.GE.I7)GO TO 182                                             RE100803
      X2=ALOG(QD(I8,1)/QD(I7,1))/(Z(I8)-Z(I7))                          RE100804
      X1=QD(I7,1)                                                       RE100805
      X3=ALOG(POX/X1)/X2+Z(I7)-Z(IPE)+DZ/2.                             RE100806
  180 WRITE(6,181)X3                                                    RE100807
  181 FORMAT(1H0,5X,24HMAXIMUM DEPTH OF DAMAGE=,E10.3,2HCM)             RE100808
C *** INTERPOLATE RADIAL EXTENT OF IRREVERSIBLE DAMAGE AT SPECIFIED     RE100809
C *** DEPTHS                                                            RE100810
  182 IF(IPRT(10).EQ.0)GO TO 192                                        RE100811
      WRITE(6,360)                                                      RE100812
  360 FORMAT(1H0,23HRADIAL EXTENT OF DAMAGE)                            RE100813
      DO 189 I=ID1,ID2                                                  RE100814
      J1=0                                                              RE100815
      X3=Z(I)-Z(IPE)+DZ/2.                                              RE100816
      DO 183 J=1,JM                                                     RE100817
      IF(POX.GT.QD(I,J))J1=J                                            RE100818
  183 CONTINUE                                                          RE100819
      X20=0.                                                            RE100820
      IF(J1.EQ.0)GO TO 187                                              RE100821
      IF(J1.EQ.JM)WRITE(6,185)X3,R(JM)                                  RE100822
  185 FORMAT(1H0,5X,2HZ=,E9.3,2HCM,5X,36HRADIAL EXTENT OF DAMAGE GREATERRE100823
     1 THAN,F10.3,2HCM)                                                 RE100824
      IF(J1.EQ.JM)GO TO 189                                             RE100825
      X2=ALOG(QD(I,J1+1)/QD(I,J1))/(R(J1+1)-R(J1))                      RE100826
      X1=QD(I,J1)                                                       RE100827
      X20=ALOG(POX/X1)/X2+R(J1)                                         RE100828
  187 WRITE(6,188)X3,X20                                                RE100829
  188 FORMAT(1H0,5X,2HZ=,E9.3,2HCM,5X,37HRADIAL EXTENT OF IRREVERSIBLE DRE100830
     1AMAGE=,E10.3,2HCM)                                                RE100831
  189 CONTINUE                                                          RE100832
      IF(LPX.EQ.0)GO TO 300                                             RE100833
      GO TO 145                                                         RE100834
  190 WRITE(6,191)                                                      RE100835
  191 FORMAT(1H0,5X,31HNO DAMAGE---LASER POWER TOO LOW)                 RE100836
  192 IF(LPX.EQ.0)GO TO 300                                             RE100837
      GO TO 145                                                         RE100838
  200 STOP                                                              RE100839
      END                                                               RE100840
      SUBROUTINE GRID                                                   RE100841
C *** GRID COMPUTES THE COEFFICIENTS IN PARTIAL DIFFERENTIAL EQUATIONS ARE100842
C *** RADIAL AND AXIAL COORDINATES, R AND Z, AND ASSIGNS CONDUCTIVITY ANRE100843
C *** VOLUMETRIC SPECIFIC HEAT TO GRID                                  RE100844
C *** CALCULATE B(CM**-2) AND R (CM)                                    RE100845
      COMMON A(29,3),AP,AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),    RE100846
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),       RE100847
     2IAB(29,14),IBLOOD(10),IFIL,IG,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,  RE100848
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2, RE100849
     4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,   RE100850
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TS(2200),TSC,TTS,V(29,14)RE100851
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),RE100852
     7XPD(120),XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,  RE100853
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC        RE100854
      DIMENSION IX(7),LX(7)                                             RE100855
```

```
C *** CALCULATE B(CM**-2) AND R (CM)
      WRITE(6,170)
  170 FORMAT(1H1)
      R(1)=0.
      CK=N-N1
      CP=RVL/DR-N1+1.
      X1=2.
  180 R2=EXP(ALOG(2.*(CP*(X1-1.)+1.)/(X1+1.))/(CK-1.))
      IF(R2/X1.GT..99999.AND.R2/X1.LT.1.00001)GO TO 181
      X1=R2
      GO TO 180
  181 IF(IPRT(1).EQ.0)GO TO 220
      WRITE(6,182)
  182 FORMAT(1H0,16HGRID INFORMATION)
      WRITE(6,184)R2
  184 FORMAT(1H0,5X,3HR2=,F8.4)
  220 RN=DR*(N1-1.+(R2**(CK+1.)-1.)/(R2-1.))
C *** CALCULATE RADIAL SPACE STEPS R(J)
      DO 185 J=2,N4
  185 R(J)=DR*(J-1)
      X1=R2*DR
      DO 186 J=N4,N
      R(J+1)=R(J)+X1
  186 X1=R2*X1
C *** CALCULATE COEFFICIENTS B OF FINITE DIFFERENCE EQNS.
      X1=2./(DR*DR)
      DO 187 J=2,N1
      B(J,1)=.25*(2*J-3)*X1/(J-1)
      B(J,2)=X1
  187 B(J,3)=X1-B(J,1)
      X2=DR
      X1=R2*DR
      DO 188 J=N4,N
      B(J,2)=2./(X1*X2)
      B(J,1)=(2./X2-1./R(J))/(X1+X2)
      B(J,3)=B(J,2)-B(J,1)
      X2=R2*X2
  188 X1=R2*X1
      B(1,1)=0.
      B(1,2)=2./(DR*DR)
      B(1,3)=B(1,2)
      DO 189 J=1,N
      IF(R(J).LT.RVL)JVL=J
  189 CONTINUE
C *** CALCULATE AXIAL SPACE STEPS Z(I)
      CK=M2-M1+1
      X1=2.
  190 CP=2.*TAV/DZ+1.-(X1**(CK-1.)-1.)/(X1-1.)
      R1=EXP(ALOG(CP*X1-CP+1.)/CK)
      IF(R1/X1.GT..99999.AND.R1/X1.LT.1.00001)GO TO 192
      X1=R1
      GO TO 190
  192 ZM=((R1**CK-1.)/(R1-1.)+M1-1.)*DZ
      IF(IPRT(1).EQ.0)GO TO 230
      WRITE(6,194)R1,ZM
  194 FORMAT(1H ,5X,3HR1=,F8.4,2X,3HZM=,F8.4)
  230 X1=DZ
```

```
      X2=X1                                                            RE100913
      DO 195 I=2,M2                                                    RE100914
      Z(M2+I)=ZM+X2                                                    RE100915
      Z(M2+2-I)=ZM-X2                                                  RE100916
      IF(I.GT.M1)X1=F1*X1                                              RE100917
  195 X2=X2+X1                                                         RE100918
      Z(1)=0.                                                          RE100919
      Z(M2+1)=ZM                                                       RE100920
      Z(M+1)=2.*ZM                                                     RE100921
      X1=Z(IPE)-DZ/2.-ZD(2)                                            RE100922
      DO 196 I=1,M3                                                    RE100923
  196 Z(I)=Z(I)-X1                                                     RE100924
      L3=IPA                                                           RE100925
      DO 200 L=1,7                                                     RE100926
      L1=0                                                             RE100927
      DO 197 I=IPA,M3                                                  RE100928
      IF(Z(I).LT.ZD(L+1))L3=I                                          RE100929
      IF(Z(I).LT.ZD(L).OR.Z(I).GE.ZD(L+1))GO TO 197                    RE100930
      L2=I                                                             RE100931
      L1=L1+1                                                          RE100932
  197 CONTINUE                                                         RE100933
      IF(L1.EQ.0)IX(L)=L3                                              RE100934
      IF(L1.EQ.0)LX(L)=L3                                              RE100935
      IF(L1.GT.0)IX(L)=L2+1-L1                                         RE100936
      IF(L1.GT.0)LX(L)=L2                                              RE100937
  200 CONTINUE                                                         RE100938
      IPV=IX(4)                                                        RE100939
      IPC=IX(5)                                                        RE100940
      IPS=IX(6)                                                        RE100941
      IPT=IX(7)                                                        RE100942
      LPA=LX(1)                                                        RE100943
      LPE=LX(3)                                                        RE100944
      LPV=LX(4)                                                        RE100945
      LPC=LX(5)                                                        RE100946
      LPS=LX(6)                                                        RE100947
      LPT=M3                                                           RE100948
C *** SET CONDUCTIVITY CON AND HEAT CAPACITY VSH FOR VARIOUS EYE MEDIA  RE100949
      DO 203 I=1,LPA                                                   RE100950
      CON(I)=CONX(1)                                                   RE100951
  203 VSH(I)=VSHX(1)                                                   RE100952
      DO 204 I=IPE,LPE                                                 RE100953
      CON(I)=CONX(2)                                                   RE100954
  204 VSH(I)=VSHX(2)                                                   RE100955
      DO 205 I=IPV,LPV                                                 RE100956
      CON(I)=CONX(3)                                                   RE100957
  205 VSH(I)=VSHX(3)                                                   RE100958
      DO 206 I=IPC,LPC                                                 RE100959
      CON(I)=CONX(4)                                                   RE100960
  206 VSH(I)=VSHX(4)                                                   RE100961
      DO 207 I=IPS,LPS                                                 RE100962
      CON(I)=CONX(5)                                                   RE100963
  207 VSH(I)=VSHX(5)                                                   RE100964
      DO 208 I=IPT,M3                                                  RE100965
      CON(I)=CONX(6)                                                   RE100966
  208 VSH(I)=VSHX(6)                                                   RE100967
C *** CALCULATE COEFFICIENTS A OF FINITE DIFFERENCE EQNS.              RE100968
      DO 210 I=IPA,M                                                   RE100969
```

```
      X1=Z(I+1)-Z(I-1)                                                  RE100970
      X2=(CON(I-1)-CON(I+1))/(X1*X1)                                    RE100971
      X3=2.*CON(I)/X1                                                   RE100972
      A(I,1)=X2+X3/(Z(I)-Z(I-1))                                        RE100973
      IF(I.EQ.IPA)A(I,1)=0.                                             RE100974
      A(I,3)=-X2+X3/(Z(I+1)-Z(I))                                       RE100975
  210 A(I,2)=A(I,1)+A(I,3)                                              RE100976
      RETURN                                                            RE100977
      END                                                               RE100978
      SUBROUTINE IMAGE                                                  RE100979
C *** IMAGE COMPUTES THE RETINAL IRRADIANCE PROFILE                     RE100980
      COMMON A(29,3),AP,AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),    RE100981
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),       RE100982
     2IAB(29,14),IBLOOD(10),IFIL,IG,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,  RE100983
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2, RE100984
     4M3,N,N1,N3,N4,NVL,POX,PP(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,   RE100985
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TS(2200),TSC,TTS,V(29,14)RE100986
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),RE100987
     7XPD(120),XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,  RE100988
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PR,PC,NB,NC,FC        RE100989
      DIMENSION FA(2001),FP(2001),FX(2001),FY(2001),JO(32),NA(22),PX(30)RE100990
     1,RX(30),XF1(2001),XF2(2001)                                       RE100991
      REAL JO,NA,NB,NC                                                  RE100992
      DO 200 J=1,N                                                      RE100993
  200 PR(J)=0.                                                          RE100994
      LI=500                                                            RE100995
      LII=LI                                                            RE100996
      DO 201 L=1,LI                                                     RE100997
  201 FX(L)=0.                                                          RE100998
      READ(5,202)PUPIL                                                  RE100999
  202 FORMAT(10E8.3)                                                    RE101000
      RINT=PUPIL/(LI-1)                                                 RE101001
      IF(IPROF.EQ.1)GO TO 214                                           RE101002
      IF(IPROF.EQ.0)GO TO 219                                           RE101003
C *** INTERPOLATE IRREGULAR LASER PROFILE(SYMMETRIC IN R) AT INTERVALS  RE101004
C *** OF RINT STARTING AT R=0                                           RE101005
      READ(5,205)LR                                                     RE101006
  205 FORMAT(I7)                                                        RE101007
      READ(5,206)(RX(L),L=1,LR)                                         RE101008
  206 FORMAT(10E7.3)                                                    RE101009
      READ(5,206)(PX(L),L=1,LR)                                         RE101010
      X1=PX(1)                                                          RE101011
      DO 207 L=1,LR                                                     RE101012
  207 PX(L)=PX(L)/X1                                                    RE101013
      X5=0.                                                             RE101014
      X6=0.                                                             RE101015
      DO 208 L=2,LR                                                     RE101016
      X2=(PX(L)-PX(L-1))/(RX(L)-RX(L-1))                                RE101017
      X1=PX(L-1)-X2*RX(L-1)                                             RE101018
      X3=X1*(RX(L)*RX(L)-RX(L-1)*RX(L-1))/2.                            RE101019
      X4=X2*(RX(L)*RX(L)*RX(L)-RX(L-1)*RX(L-1)*RX(L-1))/3.              RE101020
      IF(RX(L).GT.PUPIL)X6=X6+6.2832*(X3+X4)                            RE101021
  208 X5=X5+6.2832*(X3+X4)                                              RE101022
      QP=POX*.23906*(1.-RCO)/X5                                         RE101023
      XX=(X5-X6)/X5                                                     RE101024
      IF(RX(LR).LT.PUPIL)LII=RX(LR)/RINT+1                              RE101025
      L2=2                                                              RE101026
```

```
      X1=0.
      DO 213 L=1,LII
  210 IF(RX(L2).GT.X1)GO TO 212
      L2=L2+1
      IF(L2.LE.LR)GO TO 210
      GO TO 213
  212 X2=(X1-RX(L2-1))/(RX(L2)-RX(L2-1))
      FX(L)=PX(L2-1)+X2*(PX(L2)-PX(L2-1))
  213 X1=X1+RINT
      GO TO 223
C *** CALCULATE GAUSSIAN LASER PROFILE AT INTERVALS OF RINT STARTING AT
  214 SIGMA=RIM*SQRT(-2./ALOG(CUT))
      QP=2.*POX*.23906*(1.-RCO)/(3.1416*SIGMA*SIGMA)
      XX=1.-EXP(-2.*PUPIL*PUPIL/(SIGMA*SIGMA))
      IF(IFIL.EQ.1)GO TO 217
      DO 216 J=1,N
      X3=2.*R(J)*R(J)/(SIGMA*SIGMA)
      IF(X3.GT.80.)GO TO 216
      PR(J)=EXP(-X3)
  216 CONTINUE
      GO TO 276
  217 X1=0.
      DO 218 L=1,LII
      X3=2.*X1*X1/(SIGMA*SIGMA)
      FX(L)=0.
      IF(X3.GT.80.)GO TO 218
      FX(L)=EXP(-X3)
  218 X1=X1+RINT
      GO TO 227
C *** SPECIFY UNIFORM LASER PROFILE FROM R(1) TO R(LIM)
  219 QP=POX*.23906*(1.-RCO)/(3.1416*RIM*RIM)
      XX=1.
      IF(RIM.GT.PUPIL)XX=PUPIL*PUPIL/(RIM*RIM)
      IF(IFIL.EQ.1)GO TO 221
      DO 220 J=1,LIM
  220 PR(J)=1.
      GO TO 276
  221 L1=RIM/RINT
      RINT=RIM/L1
      LII=RIM/RINT+1
      DO 222 L=1,LII
  222 FX(L)=1.
      GO TO 227
C *** CALCULATE TOTAL AREA FA(L) AND PORTION OF LASERS POWER BETWEEN R=0
C *** AND (L-.5)*RINT
  223 IF(IFIL.EQ.1)GO TO 227
      FP(1)=3.1416*FX(1)*RINT*RINT/4.
      FA(1)=3.1416*RINT*RINT/4.
      DO 224 L=2,LII
      X1=(L-.5)*RINT
      X2=(L-1.5)*RINT
      FP(L)=FP(L-1)+FX(L)*3.1416*(X1*X1-X2*X2)
  224 FA(L)=FA(L-1)+3.1416*(X1*X1-X2*X2)
C *** CALCULATE PROFILE PR(J)
      X1=0.
      X2=0.
      DO 225 J=1,N
```

```
      X3=(R(J)+R(J+1))/(2.*RINT)+.5000001
      IF(X3.LT.1.)X3=1.000001
      L2=X3
      IF(L2.GE.LII)GO TO 225
      X4=X3-L2
      X5=FP(L2)+X4*(FP(L2+1)-FP(L2))
      X6=FA(L2)+X4*(FA(L2+1)-FA(L2))
      PR(J)=(X5-X1)/(X6-X2)
      X1=X5
      X2=X6
  225 CONTINUE
      GO TO 276
C *** SPREAD FUNCTION CALCULATIONS
  227 READ(5,202)ZO,FLO,FC,NB,CABER,PP,PC
      CABER2=CABER/WAVEL
      READ(5,228)(JO(L),L=1,32)
  228 FORMAT(10F8.5)
      READ(5,228)(NA(L),L=1,22)
      X1=(WAVEL-350.)/50.+1.
      L1=X1
      X2=X1-L1
      NC=NA(L1)+X2*(NA(L1+1)-NA(L1))
      X1=(NB-1.)*NC/(NB*(NC-1.))
      FL=FLO*X1
      X2=ZO/FLO
      X0=NC*ZO*X1/(NC*X2-X1)-FLO
      X3=1.-PC*(NC*ZO-FC)/(NC*ZO*FC)
      DO 230 L=1,LI
      IF(L.GT.LII)GO TO 230
      X1=(L-1)/X3+1.000001
      L1=X1
      X2=X1-L1
      IF(L1+1.GT.LI)FY(L)=0.
      IF(L1+1.GT.LI)LII=L
      IF(L1+1.GT.LI)GO TO 230
      FY(L)=(FX(L1)+X2*(FX(L1+1)-FX(L1)))/(X3*X3)
  230 CONTINUE
      DO 231 L=1,LII
  231 FX(L)=FY(L)
      X5=ATAN(PUPIL/(FLO-PP+X0))
      X6=1.-COS(X5)
      X7=SIN(X5)*SIN(X5)
      FF=FLO-PP
      DO 234 L=1,LII
      X4=(L-1)*RINT
      X1=6.2832*NC*(-FF-X6*X0+SQRT(FF*FF-X7*X0*X0))*X4*X4/(WAVEL*1.E-7*
     1PUPIL*PUPIL)
      X2=CABER2*X4*X4*X4*X4
      XF1(L)=SQRT(FX(L))*COS(X1+X2)
  234 XF2(L)=SQRT(FX(L))*SIN(X1+X2)
      DO 260 J=1,N
      X1=6.2832*R(J)/(WAVEL*1.E-7*FF)
      X2=0.
      X3=0.
      DO 255 L=1,LII
      X4=X1*(L-1)*RINT
      IF(L.EQ.1)X4=X1*.25*RINT
```

```
      IF(X4.GT.3.)GO TO 250                                             RE101141
      X5=X4/.1+1.000001                                                 RE101142
      L1=X5                                                             RE101143
      X5=X5-L1                                                          RE101144
      X7=JO(L1)+X5*(JO(L1+1)-JO(L1))                                    RE101145
      GO TO 251                                                         RE101146
  250 X6=3./X4                                                          RE101147
      X8=.79788456-.00000077*X6-.00552740*X6*X6-.00009512*X6*X6*X6+     RE101148
     1.00137237*X6*X6*X6*X6-.00072805*X6*X6*X6*X6*X6+.00014476*X6*X6*X6*RE101149
     2X6*X6*X6                                                          RE101150
      X9=X4-.78539816-.04166397*X6-.00003954*X6*X6+.00262573*X6*X6*X6-  RE101151
     1.00054125*X6*X6*X6*X6-.00029333*X6*X6*X6*X6*X6+.00013558*X6*X6*X6*RE101152
     2X6*X6*X6                                                          RE101153
      X7=X8*COS(X9)/SQRT(X4)                                            RE101154
  251 IF(L.GT.1)GO TO 252                                               RE101155
      X2=X2+X7*.25*(3.*XF1(1)+XF1(2))*.25*RINT*.5*RINT                  RE101156
      X3=X3+X7*.25*(3.*XF2(1)+XF2(2))*.25*RINT*.5*RINT                  RE101157
      GO TO 255                                                         RE101158
  252 X2=X2+X7*XF1(L)*(L-1)*RINT*RINT                                   RE101159
      X3=X3+X7*XF2(L)*(L-1)*RINT*RINT                                   RE101160
  255 CONTINUE                                                          RE101161
  260 HR(J)=X2*X2+X3*X3                                                 RE101162
      X1=HR(1)                                                          RE101163
      DO 270 J=1,N                                                      RE101164
  270 HR(J)=HR(J)/X1                                                    RE101165
      X1=.0002                                                          RE101166
      X2=3.1416*X1*X1/4                                                 RE101167
      J=2                                                               RE101168
      X4=HR(1)*X2                                                       RE101169
      L1=2                                                              RE101170
  271 IF(X1.LT.R(J)+.0000001)GO TO 272                                  RE101171
      J=J+1                                                             RE101172
      GO TO 271                                                         RE101173
  272 X5=(X1-R(J-1))/(R(J)-R(J-1))                                      RE101174
      X6=HR(J-1)+X5*(HR(J)-HR(J-1))                                     RE101175
      X7=8.*(L1-1)*X2                                                   RE101176
      X4=X4+X6*X7                                                       RE101177
      L1=L1+1                                                           RE101178
      X1=X1+.0002                                                       RE101179
      IF(X1.LE..1)GO TO 271                                             RE101180
      QP=.23906*XX*POX*(1.-RCO)/X4                                      RE101181
      RETURN                                                            RE101182
  276 DO 280 J=1,N                                                      RE101183
  280 HR(J)=PR(J)                                                       RE101184
      RETURN                                                            RE101185
      END                                                               RE101186
      SUBROUTINE HTXDEP                                                 RE101187
C *** HTXDEP COMPUTES RATE OF HEAT DEPOSITON AT VARIOUS POINTS I,J      RE101188
      COMMON A(29,3),AP,AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),    RE101189
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DP,DT,DTX,DZ,FL,HR(14),       RE101190
     2IAB(29,14),IBLOOD(10),IFIL,IG,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,  RE101191
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2, RE101192
     4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,   RE101193
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TS(2200),TSC,TTS,V(29,14)RE101194
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),RE101195
     7XPD(120),XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,  RE101196
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PB,PC,NB,NC,FC        RE101197
```

```
      DIMENSION AB(29,3),ABR(29,7),ABS(7),II(29),IZ(29),REF(8),REFL(8),  RE101198
     1ZH(29)                                                             RE101199
      IF(IHT.EQ.0)RETURN                                                 RE101200
      IF(QP.LT.1.E-25)GO TO 340                                          RE101201
      IF(IHT.EQ.1)RETURN                                                 RE101202
      LZ=7                                                               RE101203
      LZ0=LZ-1                                                           RE101204
      LZ1=LZ+1                                                           RE101205
      DO 280 I=1,M                                                       RE101206
      II(I)=0                                                            RE101207
      IZ(I)=0                                                            RE101208
      ZH(I)=(Z(I)+Z(I+1))/2.                                             RE101209
      DO 279 L1=1,3                                                      RE101210
  279 AB(I,L1)=0.                                                        RE101211
      DO 280 L1=1,LZ                                                     RE101212
  280 ABR(I,L1)=0.                                                       RE101213
      DO 282 L1=1,LZ                                                     RE101214
      REF(L1)=0.                                                         RE101215
  282 REFL(L1)=0.                                                        RE101216
      REF(2)=RPT                                                         RE101217
      REF(6)=RSC                                                         RE101218
      REF(LZ1)=0.                                                        RE101219
      IF(IPRT(1).EQ.0)GO TO 350                                          RE101220
      WRITE(6,283) (ZH(I),I=1,M)                                         RE101221
  283 FORMAT(1H0,5X,3HZH=/(1H ,5X,10E10.3))                              RE101222
C *** EVALUATE ABSORPTION CONSTANTS APE1 AND APE2 FOR FRONT AND REAR OF  RE101223
C *** PE AS WELL AS IG INDICATING I INDEX WHERE GRANULES ARE LOCATED     RE101224
  350 IF(IGX.EQ.1)GO TO 284                                              RE101225
      APE1=(APE-ACH*(1.-RPE))/RPE                                        RE101226
      APE2=ACH                                                           RE101227
      AP=(EXP(-ACH*RPE*TPE)-EXP(-APE1*RPE*TPE))/(1.-EXP(-APE1*RPE*TPE))  RE101228
      IG=IPE                                                             RE101229
      GO TO 285                                                          RE101230
  284 APE1=ACH                                                           RE101231
      APE2=(APE-ACH*RPE)/(1.-RPE)                                        RE101232
      AP=(EXP(-ACH*(1.-RPE)*TPE)-EXP(-APE2*(1.-RPE)*TPE))/(1.-EXP(-APE2*RE101233
     1(1.-RPE)*TPE))                                                     RE101234
      IG=LPE-(1.001-RPE)*(LPE-IPE+1)+.5                                  RE101235
  285 ABS(1)=AAV                                                         RE101236
      ABS(2)=APE1                                                        RE101237
      ABS(3)=APE2                                                        RE101238
      ABS(4)=AVL                                                         RE101239
      ABS(5)=ACH                                                         RE101240
      ABS(6)=ASC                                                         RE101241
      ABS(7)=ATS                                                         RE101242
      L1=2                                                               RE101243
      DO 306 I=IPA,M                                                     RE101244
  295 IF(ZH(I-1).LT.ZD(L1))GO TO 296                                     RE101245
      L1=L1+1                                                            RE101246
      GO TO 295                                                          RE101247
  296 IF(ZH(I).GE.ZD(L1))GO TO 299                                       RE101248
C *** NO ZD BETWEEN ZH(I-1) AND ZH(I)                                    RE101249
      AB(I,1)=ABS(L1-1)*(ZH(I)-ZH(I-1))                                  RE101250
      II(I)=1                                                            RE101251
      IZ(I)=L1                                                           RE101252
      IF(L1.GT.LZ)GO TO 306                                              RE101253
      DO 297 L2=L1,LZ                                                    RE101254
```

```
  297 ABR(I,L2)=AB(I,1)                                                 RE101255
      GO TO 306                                                         RE101256
  299 IF(ZH(I).GE.ZD(L1+1))GO TO 303                                    RE101257
C *** ONLY ZD(L1) BETWEEN ZH(I-1) AND ZH(I)                              RE101258
      AB(I,1)=ABS(L1-1)*(ZD(L1)-ZH(I-1))                                RE101259
      AB(I,2)=ABS(L1)*(ZH(I)-ZD(L1))                                    RE101260
      ABR(I,L1)=AB(I,1)                                                 RE101261
      II(I)=2                                                           RE101262
      IZ(I)=L1                                                          RE101263
      L3=L1+1                                                           RE101264
      IF(L3.GT.LZ)GO TO 306                                             RE101265
      DO 300 L2=L3,LZ                                                   RE101266
  300 ABR(I,L2)=AB(I,1)+AB(I,2)                                         RE101267
      GO TO 306                                                         RE101268
C *** ZD(L1) AND ZD(L1+1) BETWEEN ZH(I-1) AND ZH(I)                     RE101269
  303 AB(I,1)=ABS(L1-1)*(ZD(L1)-ZH(I-1))                                RE101270
      AB(I,2)=ABS(L1)*(ZD(L1+1)-ZD(L1))                                 RE101271
      AB(I,3)=ABS(L1+1)*(ZH(I)-ZD(L1+1))                                RE101272
      ABR(I,L1)=AB(I,1)                                                 RE101273
      ABR(I,L1+1)=AB(I,1)+AB(I,2)                                       RE101274
      II(I)=3                                                           RE101275
      IZ(I)=L1                                                          RE101276
      L3=L1+2                                                           RE101277
      IF(L3.GT.LZ)GO TO 306                                             RE101278
      DO 304 L2=L3,LZ                                                   RE101279
  304 ABR(I,L2)=AB(I,1)+AB(I,2)+AB(I,3)                                 RE101280
  306 CONTINUE                                                          RE101281
      DO 314 I=IPA,M                                                    RE101282
      IF(AB(I,1).GT.10.)AB(I,1)=10.                                     RE101283
      IF(AB(I,2).GT.10.)AB(I,2)=10.                                     RE101284
      IF(AB(I,3).GT.10.)AB(I,3)=10.                                     RE101285
      DO 314 L=2,LZ                                                     RE101286
      IF(ABR(I,L).GT.10.)ABR(I,L)=10.                                   RE101287
  314 CONTINUE                                                          RE101288
C *** DEPOSITION BY INCOMING BEAM                                       RE101289
      X2=QP                                                             RE101290
      L1=2                                                              RE101291
      DO 317 I=IPA,M                                                    RE101292
      L2=II(I)                                                          RE101293
      X3=X2                                                             RE101294
      X2=X2*EXP(-AB(I,1))                                               RE101295
      X4=0.                                                             RE101296
      IF(L2.EQ.1)GO TO 315                                              RE101297
      L3=IZ(I)                                                          RE101298
      X4=X2*REF(L3)                                                     RE101299
      X2=X2*(1.-REF(L3))*EXP(-AB(I,2))                                  RE101300
      IF(L2.EQ.2)GO TO 315                                              RE101301
      X4=X4+X2*REF(L3+1)                                                RE101302
      X2=X2*(1.-REF(L3+1))*EXP(-AB(I,3))                                RE101303
  315 IF(X2.LT.1.E-10)X2=0.                                             RE101304
      DO 317 J=1,JVL                                                    RE101305
      S(I,J)=(X3-X2-X4)*HP(J)/(ZH(I)-ZH(I-1))                           RE101306
      IF(S(I,J).LT.1.E-10/DPULSE)S(I,J)=0.                              RE101307
  317 CONTINUE                                                          RE101308
C *** CALCULATION OF REFLECTED INTENSITIES BY VARIOUS INTERFACES        RE101309
C *** STARTING WITH FIRST INTERNAL INTERFACE                            RE101310
      X2=QP                                                             RE101311
```

```
      DO 322 L1=1,LZO
      X3=ABS(L1)*(ZD(L1+1)-ZD(L1))
      IF(X3.GT.10.)X3=10.
      X2=X2*EXP(-X3)
      REFL(L1+1)=X2*REF(L1+1)
  322 X2=X2*(1.-REF(L1+1))
      DO 327 L1=2,LZ
      I=IPA
  324 IF(ZH(I).GT.ZD(L1))GO TO 325
      I=I+1
      IF(I.LE.M)GO TO 324
      GO TO 327
  325 X2=REFL(L1)
      DO 326 L3=IPA,I
      X3=X2
      L4=I+IPA-L3
      X2=X2*EXP(-ABR(L4,L1))
      DO 326 J=1,JVL
      S(L4,J)=S(L4,J)+(X3-X2)*HR(J)/(ZH(L4)-ZH(L4-1))
      IF(S(L4,J).LT.1.E-10/DPULSE)S(L4,J)=0.
  326 CONTINUE
  327 CONTINUE
      IHT=1
      RETURN
C *** NO HEAT DEPOSITION,BEAM OFF
  340 DO 342 I=1,M3
      DO 342 J=1,N3
  342 S(I,J)=0.
      IHT=0
      RETURN
      END
      SUBROUTINE MXGRAN
C *** THIS ROUTINE COMPUTES CONSEQUENCE OF GRANULAR ABSORPTION ON
C *** TEMPERATURE VARIATIONS IN PE. (USED ONLY ONCE.)
      COMMON A(29,3),AP,AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),
     2IAB(29,14),IBLOOD(10),IFIL,IG,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2,
     4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TS(2200),TSC,TTS,V(29,14)
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),
     7XPD(120),XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC
      L5=1
      BT=.3E-8
      IF(IPRT(7).EQ.0)GO TO 407
      WRITE(6,403)
  403 FORMAT(1H0,48HNORMALIZED TEMPERATURE RISES OF MELANIN GRANULES)
      WRITE(6,405)LTMAX,BT
  405 FORMAT(1H0,5X,6HLTMAX=,I4,2X,3HBT=,E8.3)
  407 IF(DPULSE.GT.1.0E-5)GO TO 494
      LPT=DPULSE/.3E-8
      L7=LTMAX-10
      DO 410 L=1,L7,10
      L1=L+1
      L2=L+9
      X1=TS(L)
```

```
      X2=TS(L+10)-X1
      X3=0.
      DO 410 L3=L1,L2
      X3=X3+.1
  410 TS(L3)=X1+X3*X2
      LTT=2
      XPD(1)=1.0
      IF(LPT.GE.LTMAX)GO TO 472
C *** CASE FOR LPT LESS THAN LTMAX------
  440 IF(XT(LTT).GE..3E-8)GO TO 442
      XPD(LTT)=TS(1)
      LTT=LTT+1
      GO TO 440
  442 TIMEX=XT(LTT)
      XX=TIMEX/BT+.000001
      LX=XX
      PT=LX
      IF(LX.GT.LPT)PT=LPT
      PO=0.
      IF(LX.GT.LTMAX)PO=LX-LTMAX
      IF(LX.GT.LPT)GO TO 443
      L1=1
      L2=LX
      GO TO 446
  443 IF(LX.GT.LTMAX)GO TO 444
      L1=LX+1-LPT
      L2=LX
      GO TO 446
  444 IF(LX.LT.LTMAX+LPT)GO TO 445
      L5=LTT
      GO TO 494
  445 L1=LX-LPT+1
      L2=LTMAX
  446 X2=PO
      DO 448 L3=L1,L2
  448 X2=X2+TS(L3)
      XPD(LTT)=X2/PT
      LTT=LTT+1
      IF(LTT.LE.KT)GO TO 442
      GO TO 496
C *** CASE FOR LTMAX LESS THAN LPT------
  472 TIMEX=XT(LTT)
      XX=TIMEX/BT+.000001
      LX=XX
      PT=LX
      IF(LX.GT.LPT)PT=LPT
      PO=0.
      IF(LX.GT.LTMAX)PO=LX-LTMAX
      IF(LX.GT.LPT)GO TO 473
      L1=1
      L2=LX
      IF(LX.GT.LTMAX)L2=LTMAX
      GO TO 475
  473 IF(LX.LT.LTMAX+LPT)GO TO 474
      L5=LTT
      GO TO 494
  474 L1=LX-LPT+1
```

```
      L2=LTMAX                                                          RE101426
  475 X2=PO                                                             RE101427
      DO 476 L3=L1,L2                                                   RE101428
  476 X2=X2+TS(L3)                                                      RE101429
      XPD(LTT)=X2/PT                                                    RE101430
      LTT=LTT+1                                                         RE101431
      IF(LTT.LE.KT)GO TO 472                                            RE101432
      GO TO 496                                                         RE101433
C *** END CALCULATION IF TEMPERATURES VERY UNIFORM                       RE101434
  494 IF(L5.GT.KT)GO TO 496                                             RE101435
      DO 495 L1=L5,KT                                                   RE101436
  495 XPD(L1)=1.                                                        RE101437
  496 IF(IPRT(7).EQ.0)RETURN                                            RE101438
      WRITE(6,497)(XPD(L1),L1=1,KT)                                     RE101439
  497 FORMAT(1H0,5X,4HXPD=/(1H ,5X,10F8.2))                             RE101440
      RETURN                                                            RE101441
      END                                                               RE101442
      SUBROUTINE BLOOD                                                  RE101443
C     SUBROUTINE BLOOD COMPUTES CHANGES IN MATRIX ELEMENTS A AND B DUE  RE101444
C     TO BLOOD FLOW                                                     RE101445
      COMMON A(29,3),AP,AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),    RE101446
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),       RE101447
     2IAB(29,14),IBLOOD(10),IFIL,IG,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,  RE101448
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,LTMAX,K,KM,KT,M,M1,M2, RE101449
     4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,   RE101450
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TS(2200),TSC,TTS,V(29,14)RE101451
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),RE101452
     7XPD(120),XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,  RE101453
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC        RE101454
      DIMENSION RD(14),RH(14),XI(14),XO(14)                             RE101455
C *** INITIAL EVALUATION OF PARAMETERS AND ARRAYS                       RE101456
      DO 800 J=1,N3                                                     RE101457
      BV(J,1)=0.                                                        RE101458
      BV(J,2)=0.                                                        RE101459
      BV(J,3)=0.                                                        RE101460
      FLOWI(J)=0.                                                       RE101461
  800 FLOWX(J)=0.                                                       RE101462
      RH(1)=R(2)/2.                                                     RE101463
      DO 803 J=2,JVL                                                    RE101464
  803 RH(J)=(R(J)+R(J+1))/2.                                            RE101465
      L2=2                                                              RE101466
      DO 810 J=1,JVL                                                    RE101467
  805 IF(DFLOW(L2).GT.RH(J))GO TO 806                                   RE101468
      L2=L2+1                                                           RE101469
      GO TO 805                                                         RE101470
  806 X1=DFLOW(L2)-DFLOW(L2-1)                                          RE101471
      X2=RH(J)-DFLOW(L2-1)                                              RE101472
      X3=X2/X1                                                          RE101473
      XI(J)= XFLOWI(L2-1)+X3*(XFLOWI(L2)-XFLOWI(L2-1))                  RE101474
  810 XO(J)= XFLOWO(L2-1)+X3*(XFLOWO(L2)-XFLOWO(L2-1))                  RE101475
      FLOWX(1)=0.                                                       RE101476
      DO 812 J=2,JVL                                                    RE101477
  812 FLOWX(J)=FLOWX(J-1)+(XI(J-1)-XO(J-1))*(R(J)*R(J)-R(J-1)*R(J-1))/  RE101478
     1(2.*TVL)                                                          RE101479
      FLOWX(JVL+1)=FLOWX(JVL)                                           RE101480
      L2=2                                                              RE101481
      FLOWI(1)=XFLOWI(1)/TVL                                            RE101482
```

```
      DO 820 J=2,JVL
  814 IF(DFLOW(L2).GT.R(J))GO TO 816
      L2=L2+1
      GO TO 814
  816 X4=DFLOW(L2)-DFLOW(L2-1)
      X5=R(J)-DFLOW(L2-1)
      X6=X5/X4
  820 FLOWI(J)=(XFLOWI(L2-1)+X6*(XFLOWI(L2)-XFLOWI(L2-1)))/TVL
      DO 823 J=2,JVL
  823 RD(J)=1./(R(J)*(R(J+1)-R(J-1)))
C *** CALCULATE CHANGES IN MATRIX ELEMENTS A AND B DUE TO BLOOD FLOW
      BV(1,1)=0.
      BV(1,2)=-SHB*FLOWI(1)/2.
      BV(1,3)=0.
      BB=-SHB*XFLOW/2.
      DO 825 J=2,JVL
      BV(J,1)=SHB*RD(J)*FLOWX(J)
      BV(J,2)=SHB*RD(J)*(FLOWX(J-1)-FLOWX(J+1))/2.-SHB*FLOWI(J)/2.
  825 BV(J,3)=-SHB*RD(J)*FLOWX(J)
      DO 835 I=IPA,M
  835 IV(I)=0
      DO 840 L3=1,NVL
      L4=IBLOOD(L3)
  840 IV(L4)=1
      DO 845 I=IPA,LPS
      DO 842 J=1,JVL
  842 IAB(I,J)=0
      IF(JVL.EQ.N)GO TO 845
      L1=JVL+1
      DO 843 J=L1,N
  843 IAB(I,J)=1
  845 CONTINUE
      DO 850 I=IPT,M
      DO 850 J=1,N
  850 IAB(I,J)=1
      RETURN
      END
```

1519 RECORDS PRINTED

```
C                          RETINAL MODEL  IITRI
C                          VERSION 2  14 NOV 1975
C   TEMPERATURE AND DAMAGE PREDICTIONS IN AND ABOUT RET CAUSED BY LASERS
C   EFFECTS OF MELANIN GRANULES NOT INCORPORATED IN PROGRAM
C
      COMMON A(29,3),AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HP(14),
     2IAB(29,14),IBLOOD(10),IFIL,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,K,KM,KT,M,M1,M2,
     4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TSC,TTS,V(29,14)
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),
     7XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC
      DIMENSION CXC(14),CXR(29),DAMAGE(2,2),DXC(14),DXR(29),FTIME(38),
     1FXC(14),FXR(29),ID(230),JD(230),KTT(38),NPT(38),NPULSE(7),NRUN(7),
     2QD(29,14),REPET(7),TIMEX(10),XCT(38),XQD(29,14),VE(27,120,1),
     3VXX(29,14),VZ(27,42,8,1),ZT(8),ZTT(8),SAVRGV(10)
      REAL LESION
    2 FORMAT(10F7.3)
    3 FORMAT(F7.4,3I7)
    4 FORMAT(11F7.2)
    5 FORMAT(10I7)
    6 FORMAT(F7.2,I7,2F7.2)
    7 FORMAT(10E7.2)
    8 FORMAT(I7,3E7.2)
    9 FORMAT(F7.2,2I7,F7.2)
  300 READ(5,4,END=200) (FTIME(L),L=1,38)
      READ(5,5) IPRT
      READ(5,3) RIM,LIM,IFIL,IGX
      READ(5,9) RMAX,LIMAX,MAXPRT,LESION
C *** SET VALUES FOR MTEST,N,N1,N3,N4, AND DR
      MTEST=0
      N1=4
      N=N1+9
      N3=N+1
      N4=N1+1
      READ(5,8) IPROF,POW,CUT
      DR=LESION/LIM
      IF(IPROF.EQ.0) DR=RIM/(LIM-.5)
      READ(5,7) DPULSE
      READ(5,5) NTEST,(NRUN(L),L=1,NTEST)
      READ(5, 7) (REPET(L),L=1,NTEST)
      READ(5, 5) (NPULSE(L),L=1,NTEST)
      READ(5,5) ID1,ID2,JD1,JD2,ITYPE
      LPX=1
      IF(NTEST.EQ.1.AND.NPULSE(1).EQ.1) LPX=0
      XDPULS=DPULSE
      XXQ=1.
      IF(DPULSE.GT..3E-8) GO TO 10
C *** ADJUST POWER AND PULSE WIDTH FOR EXPOSURES WITH PULSES LESS THAN
C *** .3E-8 SEC
      XXQ=.3E-8/DPULSE
      POW=POW*DPULSE/.3E-8
      DPULSE=.3E-8
   10 READ(5,4) TO,EDT1,EDT2
      READ(5,4) TOM,APE,AVL,ACH,ASC,ATS,RCO,RRT,RSC,RPE,WAVEL
```

```
      READ(5,4)TAV,TPE,TVL,TCH,TSC,RVL
      AAV=-ALOG(TOM)/TAV
      READ(5, 4) (CONX(L),L=1,6)
      READ(5,4) (VSHX(L),L=1,6)
      READ(5,5) (NPT(L),L=1,38)
      READ(5,2) (XCT(L),L=1,38)
      READ(5,5) (KTT(L),L=1,38)
C *** COMPUTE DT,KM,KT,NP,PTIME,TIME, AND XC
      L1=ALOG(DPULSE)/.69315+29.
      IF(L1.LT.1)L1=1
      IF(L1.GT.38)L1=38
      IF(LPX.EQ.1)GO TO 11
C *** ---SINGLE PULSED EXPOSURES
      XC=XCT(L1)
      NP=NPT(L1)
      KT=KTT(L1)
      DT=DPULSE*(XC-1.)/(XC**NP-1.)
      TIME=DT*(XC**KT-1.)/(XC-1.)
      GO TO 13
C *** ---MULTIPLE PULSED EXPOSURES
   11 XC=1.4
      NP=5
      X1=0.
      DO 12 L=1,NTEST
      IF(X1.LT.NPULSE(L)/REPET(L))X1=NPULSE(L)/REPET(L)
   12 CONTINUE
      TIME=FTIME(L1)*X1
      DT=DPULSE*(XC-1.)/(XC**NP-1.)
      KT=ALOG(1.+TIME*(XC-1.)/DT)/ALOG(XC)+1.
      PTIME=DPULSE/NP
   13 KT=KT+1
      KM=NP+1
      IF(KT.GT.119)WRITE(6,14)KT
   14 FORMAT(1H0,3HKT=,I3,2X,22HTIME DIMENSION TOO LOW)
      IF(KT.GT.119)STOP
C *** CALC. DZ AND I INDICES
      M1=6
      M=2*M1+16
      M2=M/2
      M3=M+1
      IPE=M2-M1+2
      DZ=TPE/M1-1.E-25
      IPA=2
C *** STORE AXIAL DISTANCES TO INTERFACES OF EYE
      ZD(1)=1.E-25
      ZD(2)=TAV
      ZD(3)=ZD(2)+RPE*TPE
      ZD(4)=ZD(3)+(1.-RPE)*TPE
      ZD(5)=ZD(4)+TVL
      ZD(6)=ZD(5)+TCH
      ZD(7)=ZD(6)+TSC
      ZD(8)=ZD(7)+10.
      CALL GRID
      NVL=LPV-IPV+1
C *** CALCULATE AND STORE I,J INDICES AT WHICH TEMPERATURES ARE PRINTED
      ID1=ID1+IPE
      ID2=ID2+IPE
```

```
      IF(ID1.LT.IPA)ID1=IPA
      IF(ID2.GT.M)ID2=M
      IF(JD2.GT.N)JD2=N
      IF(IPRT(1).EQ.0)GO TO 23
      WRITE(6,15)ID1,ID2,JD1,JD2
   15 FORMAT(1H0,5X,4HID1=,I3,3X,4HID2=,I3,3X,4HJD1=,I2,3X,4HJD2=,I2)
      WRITE(6,16)DR,DZ
   16 FORMAT(1H0,5X,3HDR=,E11.4,2X,3HDZ=,E11.4)
      WRITE(6,17)IPA,IPC,IPE,IPS,IPT,IPV,LPA,LPC,LPE,LPS,LPV
   17 FORMAT(1H0,5X,4HIPA=,I3,2X,4HIPC=,I3,2X,4HIPE=,I3,2X,4HIPS=,I3,2X,
     14HIPT=,I3,2X,4HIPV=,I3/1H ,5X,4HLPA=,I3,2X,4HLPC=,I3,2X,4HLPE=,I3,
     22X,4HLPS=,I3,2X,4HLPV=,I3)
      WRITE(6,22)M,M1,N,N1
   22 FORMAT(1H0,5X,2HM=,I2,2X,3HM1=,I2,2X,2HN=,I2,2X,3HN1=,I2)
      WRITE(6,18) (R(J),J=1,N3)
   18 FORMAT(1H0,5X,2HR=/(1H ,5X,10F8.4))
      WRITE(6,19) (Z(I),I=1,M3)
   19 FORMAT(1H0,5X,2HZ=/(1H ,5X,10F8.4))
   23 DO 20 L1=1,NVL
   20 IBLOOD(L1)=IPV+L1-1
C *** CALC. NORMALIZED LASER PROFILES---
      DO 21 L=1,N3
   21 HR(L)=0.
      POX=POW
      CALL IMAGE
      DO 27 J=1,N3
      DO 27 I=1,M3
      V(I,J)=1.E-10
   27 S(I,J)=0.
      READ(5,2)SHB,XFLOW,CFLOW
C *** SET BLOOD FLOW RATES ENTERING AND LEAVING VASCULAR LAYER AS
C *** FUNCTION OF RADIAL DISTANCE
      X2=CFLOW/(3.1416*RVL*RVL)
      DFLOW(1)=0.
      X4=0.
      DO 30 L1=2,6
      X4=X4+.1
   30 DFLOW(L1)=X4
      DO 31 L1=1,6
      XFLOWI(L1)=X2
   31 XFLOWO(L1)=X2
      DO 34 I=1,M3
      DO 34 J=1,N3
   34 VC(I,J,1)=1.E-10
      XPOW=XXQ*POW
      READ(5,8)KTYPEO
      READ(5,8)KTYPE
      L1=KTYPE
      IF(KTYPE.EQ.0)L1=1
      READ(5,7) (TIMEX(K),K=1,L1)
      READ(5,5)II1,II2,II3,JJ1,JJ2
C *** START OF TEMPERATURE CALCULATIONS FOR ONE PULSE. TO BE USED EITHER
C *** FOR MULTIPLE OR SINGLE PULSED EXPOSURES
C     ---------------------------------------------------------------
      XT(1)=0.
      DTX=DT
      KTX=KT+1
```

```
      DO 36 K=2,KTX
      XT(K)=XT(K-1)+DT
   36 DT=XC*DT
      IKX=TIME**EDT1+EDT2
      IF(IKX.LT.1)IKX=1
      XX=2*IKX
      K=2
      IHT=2
      ITYPEX=ITYPE
      CALL BLOOD
   38 DT=XT(K)-XT(K-1)
      IF(K.GT.KM)QP=0.
      CALL HTXDEP
      IF(K.GT.2)GO TO 41
      IF(IPRT(2).EQ.0)GO TO 335
      WRITE(6,301)
  301 FORMAT(1H0,13HLASER PROFILE)
      IF(IPROF.EQ.0)WRITE(6,302)RIM
  302 FORMAT(1H0,5X,4HRIM=,E10.3)
      IF(IPROF.EQ.1)WRITE(6,303)SIGMA,RIM,CUT
  303 FORMAT(1H0,5X,6HSIGMA=,E10.3,5X,4HRIM=,E10.3,5X,4HCUT=,E10.3)
      IF(IFIL.EQ.1)WRITE(6,304)RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC,
     1WAVEL
  304 FORMAT(1H0,5X,5HRINT=,E10.3,3X,3HZO=,E10.3,3X,4HFLO=,F6.3/1H ,5X,
     16HCABER=,E10.3,3X,7HCABER2=,F7.0,3X,3HPP=,F6.3/1H ,5X,3HPC=,F6.3,
     23X,3HNB=,F7.3,3X,3HNC=,F7.3/1H ,5X,3HFC=,F6.3,3X,6HWAVEL=,F7.1)
      IF(IFIL.EQ.1)GO TO 306
      IF(IPROF.EQ.2)WRITE(6,305)RINT
  305 FORMAT(1H0,5X,5HRINT=,E10.3)
  306 WRITE(6,307)QP
  307 FORMAT(1H0,5X,3HQP=,E10.3)
      WRITE(6,308)(HR(J),J=1,N)
  308 FORMAT(1H0,5X,3HHR=/(1H ,10X,10E10.3))
  335 IF(IPRT(3).EQ.0)GO TO 340
      WRITE(6,309)
  309 FORMAT(1H0,19HDATA IDENTIFICATION)
      WRITE(6,310)(REPET(L),L=1,NTEST)
  310 FORMAT(1H0,5X,6HREPET=/(1H ,5X,10E10.3))
      WRITE(6,311)(NPULSE(L),L=1,NTEST)
  311 FORMAT(1H0,5X,7HNPULSE=/(1H ,5X,10I8))
      WRITE(6,312)AAV,ACH,APE,ASC,ATS,RCO,RRT,RPE,TOM,AVL,TAV,TCH,TPE,
     1TSC,TVL,IGX,IFIL,IPROF,LIM,NTEST,POW,XDPULS,RIM,RMAX,TIME,CFLOW,
     2XFLOW,SHB,EDT1,EDT2,DT,KM,KT,PTIME,XC,IKX,APE1,APE2,RVL,
     3PUPIL,TO,LIMAX,MAXPRT
  312 FORMAT(1H0,5X, 4HAAV=,F7.1,2X,4HACH=,F7.0,2X,4HAPE=,F7.0,2X,4HASC=
     1,F7.0,2X,4HATS=,F7.0/1H ,5X,4HRCO=,F7.4,2X,4HRRT=,F7.4,2X,4HRPE=,
     2F7.4,2X,4HTOM=,F7.4,2X,4HAVL=,F7.0/1H ,5X,4HTAV=,E9.3,2X,4HTCH=,
     3E9.3,2X,4HTPE=,E9.3,2X,4HTSC=,E9.3,2X,4HTVL=,E9.3/1H ,5X,4HIGX=,I2
     4,2X,5HIFIL=,I2,2X,6HIPROF=,I2,2X,4HLIM=,I2,2X,6HNTEST=,I2/1H ,5X,
     54HPOW=,E9.3,2X,7HDPULSE=,E9.3,2X,4HRIM=,F7.4,2X,5HRMAX=,F7.4,2X,
     65HTIME=,E9.3/1H ,5X,6HCFLOW=,F7.4,2X,6HXFLOW=,F7.4,2X,4HSHB=,F7.2,
     72X,5HEDT1=,F7.4,2X,5HEDT2=,F7.4/1H ,5X,3HDT=,E9.3,2X,3HKM=,I3,2X,
     83HKT=,I3,2X,6HPTIME=,E9.3,2X,3HXC=,F5.1/1H ,5X,4HIKX=,I2,2X,
     9          5HAPE1=,F8.2,2X,5HAPE2=,F8.2              /1H ,5X,4HRVL=,
     1F6.3,2X,6HPUPIL=,F6.3,2X,3HTO=,F5.1,2X,6HLIMAX=,I2,2X,7HMAXPRT=,
     2I2)
  340 IF(IPRT(4).EQ.0)GO TO 355
```

```
      WRITE(6,313)
  313 FORMAT(1H0,30HBLOOD FLOW AND HEAT DEPOSITION)
      WRITE(6,314) (FLOWI(J),J=1,JVL)
  314 FORMAT(1H0,5X,6HFLOWI=/(1H ,5X,10E10.3))
      WRITE(6,315) (FLOWX(J),J=1,JVL)
  315 FORMAT(1H0,5X,6HFLOWX=/(1H ,5X,10E10.3))
      WRITE(6,316)
  316 FORMAT(1H )
      DO 318 I=IPA,M
      WRITE(6,317) (S(I,J),J=1,N)
  317 FORMAT(1H ,5X,2HS=,10E8.3)
  318 CONTINUE
  355 IF(IPRT(5).EQ.0)GO TO 41
      WRITE(6,319)
  319 FORMAT(1H0,17HTEMPERATURE RISES)
      JCNT=JD2-JD1+1
      IF(JCNT.GT.9)GO TO 40
      GO TO 41
   40 JJCNT=JCNT-9
      JJD2=JD2-JJCNT
      JJD2P1=JJD2+1
   41 IF(IPRT(5).EQ.0)GO TO 356
      WRITE(6,42)XT(K),K
   42 FORMAT(1H0,5X,5HTIME=,E11.4,3X,2HK=,I3)
C *** CALCULATE TEMPERATURE RISE(MATRIX REDUCTION ALGORITHM)
C *** COLUMNS(NORMAL)-------------------
  356 IK=1
   43 DO 45 I=IPA,M
      W=XX*VSH(I)/DT
      DO 44 J=1,N
      FXC(J)=W+CON(I)*B(J,2)-BV(J,2)*IV(I)-BB*IAB(I,J)
      IF(J.GT.1)FXC(J)=FXC(J)+(CON(I)*B(J,1)+BV(J,1)*IV(I))*CXC(J-1)
      CXC(J)=-(CON(I)*B(J,3)+BV(J,3)*IV(I))/FXC(J)
      SUM=(W-(A(I,2)-BV(J,2)*IV(I)-BB*IAB(I,J)))*V(I,J)+A(I,1)*V(I-1,J)+
     1A(I,3)*V(I+1,J)+S(I,J)
      DXC(J)=SUM/FXC(J)
      IF(J.GT.1)DXC(J)=(SUM+(CON(I)*B(J,1)+BV(J,1)*IV(I))*DXC(J-1))/FXC(
     1J)
   44 CONTINUE
      VX=0.
      DO 45 L=1,N
      J=N+1-L
      VX=DXC(J)-CXC(J)*VX
   45 VXX(I,J)=VX
      DO 46 I=IPA,M
      DO 46 J=1,N
   46 V(I,J)=VXX(I,J)
C *** ROWS(NORMAL)----------------------
      CXR(IPA-1)=0.
      DO 50 J=1,N
      DO 48 I=IPA,M
      W=XX*VSH(I)/DT
      FXR(I)=W+A(I,2)-BV(J,2)*IV(I)-BB*IAB(I,J)+A(I,1)*CXR(I-1)
      CXR(I)=-A(I,3)/FXR(I)
      SUM=(W-(CON(I)*B(J,2)-BV(J,2)*IV(I)-BB*IAB(I,J)))*V(I,J)+(CON(I)*
     1B(J,3)+BV(J,3)*IV(I))*V(I,J+1)+S(I,J)
      IF(J.GT.1)SUM=SUM+(CON(I)*B(J,1)+BV(J,1)*IV(I))*V(I,J-1)
```

```
      DXR(I)=SUM/FXR(I)
      IF(I.GT.IPA)DXR(I)=(SUM+A(I,1)*DXR(I-1))/FXR(I)
   48 CONTINUE
      VX=0.
      DO 50 L=IPA,M
      I=M+IPA-L
      VX=DXR(I)-CXR(I)*VX
      VC(I,J,K)=VX
   50 VXX(I,J)=VX
      DO 51 I=IPA,M
      DO 51 J=1,N
   51 V(I,J)=VXX(I,J)
      IK=IK+1
C *** RECYCLE TEMPERATURE CALCULATIONS
      IF(IK.LE.IKX)GO TO 43
      IF(K.EQ.KM)GO TO 62
      IF(ITYPEX.LT.ITYPE.AND.K.LT.KT)GO TO 66
   62 IF(IPRT(5).EQ.0)GO TO 357
      WRITE(6,63) (R(J),J=JD1,JD2)
   63 FORMAT(1H ,13X,2HR=,9F13.5/1H ,15X,30H----------------------------
     1--)
      DO 65  I=ID1,ID2
      X1=Z(I)-Z(IPE)+DZ/2.
      IF(JCNT.GT.9)GO TO 57
      WRITE(6,64)X1,(VC(I,J,K),J=JD1,JD2)
      GO TO 65
   57 WRITE(6,64)X1,(VC(I,J,K),J=JD1,JJD2)
      WRITE(6,64)X1,(VC(I,J,K),J=JJD2P1,JD2)
   64 FORMAT(1H ,3X,2HZ=,F8.5,2X,1P9E13.6)
   65 CONTINUE
  357 ITYPEX=0
   66 K=K+1
      ITYPEX=ITYPEX+1
      IF(K.LE.KT)GO TO 38
      ITYPEX=ITYPE
      IF(IPRT(6).EQ.0)GO TO 365
      WRITE(6,320)
  320 FORMAT(1H0,28HNORMALIZED TEMPERATURE RISES)
      DO 70 K=2,KT
      IF(K.EQ.KM)GO TO 67
      IF(ITYPEX.LT.ITYPE.AND.K.LT.KT)GO TO 70
   67 X1=1.
      WRITE(6,321)XT(K),K,X1
  321 FORMAT(1H0,5X,5HTIME=,E11.4,3X,2HK=,I3,3X,6HPOWER=,E11.4,5HWATTS)
      WRITE(6,63)(R(J),J=JD1,JD2)
      JCNT=JD2-JD1+1
      IF(JCNT.GT.9)GO TO 380
      GO TO 381
 380  JJCNT=JCNT-9
      JJD2=JD2-JJCNT
      JJD2P1=JJD2+1
 381  DO 69 I=ID1,ID2
      DO 68 J=JD1,JD2
   68 V(I,J)=VC(I,J,K)/POW
      X1=Z(I)-Z(IPE)+DZ/2.
      IF(JCNT.GT.9)GO TO 382
      WRITE(6,64)X1,(V(I,J),J=JD1,JD2)
```

```
      GO TO 69                                                          RE200343
 382  WRITE(6,64)X1,(V(I,J),J=JD1,JJD2)                                 RE200344
      WRITE(6,64)X1,(V(I,J),J=JJD2P1,JD2)                               RE200345
  69  CONTINUE                                                          RE200346
      ITYPEX=0                                                          RE200347
  70  ITYPEX=ITYPEX+1                                                   RE200348
 330  FORMAT(1H0,61HDIMENSION OF ARRAYS ASSOCIATED WITH ARGUMENT LIJ IS RE200349
     1TOO SMALL)                                                        RE200350
 365  CONTINUE                                                          RE200351
      READ(5,4)(DAMAGE(L2,1),DAMAGE(L2,2),L2=1,2)                       RE200352
      WRITE(6,73)WAVEL,DAMAGE(1,1),DAMAGE(1,2),DAMAGE(2,1),             RE200353
     1DAMAGE(2,2)                                                       RE200354
  73  FORMAT(1H0,5X,11HWAVELENGTH=,F7.1,2HNM,3X,7HDAMAGE=,4F9.0)        RE200355
C *** CALCULATE I,J INDICES AT WHICH DAMAGE CALCULATIONS ARE TO BE MADE RE200356
      JM=0                                                              RE200357
      DO 74 J=1,N                                                       RE200358
      IF(R(J).LT.RMAX+.000001)JM=J+1                                    RE200359
  74  CONTINUE                                                          RE200360
      X1=0.                                                             RE200361
      DO 75 I=IPA,M                                                     RE200362
      IF(VC(I,1,KM).GT.X1)IMAX=I                                        RE200363
      IF(VC(I,1,KM).GT.X1)X1=VC(I,1,KM)                                 RE200364
  75  CONTINUE                                                          RE200365
      L=0                                                               RE200366
      GO TO (366,367,368),MAXPRT                                        RE200367
 366  LIMAX1=2*LIMAX                                                    RE200368
      LIMAX2=0                                                          RE200369
      GO TO 369                                                         RE200370
 367  LIMAX1=LIMAX                                                      RE200371
      LIMAX2=LIMAX                                                      RE200372
      GO TO 369                                                         RE200373
 368  LIMAX1=0                                                          RE200374
      LIMAX2=2*LIMAX                                                    RE200375
 369  ID1=IMAX-LIMAX1                                                   RE200376
      ID2=IMAX+LIMAX2                                                   RE200377
      IF(ID2.GT.28)ID2=28                                               RE200378
      DO 76 I=ID1,ID2                                                   RE200379
      DO 76 J=1,JM                                                      RE200380
      L=L+1                                                             RE200381
      ID(L)=I                                                           RE200382
  76  JD(L)=J                                                           RE200383
      LIJ=(ID2-ID1+1)*JM                                                RE200384
      DO 505 LL15=1,10                                                  RE200385
 505  SAVRGV(LL15)=0.                                                   RE200386
      IF(LPX.EQ.0)GO TO 125                                             RE200387
      IF(LIJ.GT.27)WRITE(6,330)                                         RE200388
      IF(LIJ.GT.27)GO TO 300                                            RE200389
      IF(IPRT(8).EQ.0)GO TO 370                                         RE200390
C *** TEMPERATURE AND DAMAGE EVALUATIONS FOR MULTIPLE PULSES            RE200391
C     -----------------------------------------------------------       RE200392
C *** EVALUATE TEMPERATURE RISES WITHOUT GRANULES                       RE200393
      DO 77 L=1,LIJ                                                     RE200394
      I=ID(L)                                                           RE200395
      J=JD(L)                                                           RE200396
      VE(L,1,1)=0.                                                      RE200397
      DO 77 K=2,KT                                                      RE200398
      VE(L,K,1)=VC(I,J,K)                                               RE200399
```

AD-A034 239 SCHOOL OF AEROSPACE MEDICINE BROOKS AFB TEX F/G 9/2

RETINAL THERMAL MODEL OF LASER-INDUCED EYE DAMAGE: COMPUTER OPE--ETC(U)

NOV 76 A R MERTZ, B R ANDERSON, E L BELL

UNCLASSIFIED SAM-TR-76-33 NL

2 OF 2

AD

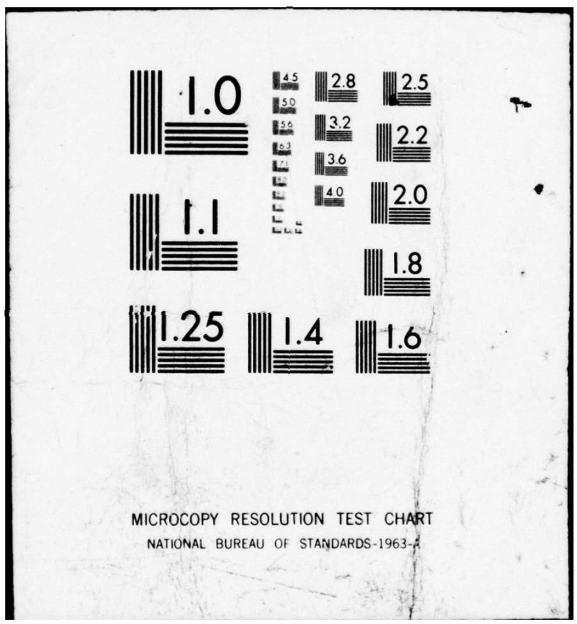
A034239

END

DATE

FILMED

2-77


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
   77 CONTINUE                                                           RE200400
      X60=(XC-1.)/DTX                                                    RE200401
      X61=ALOG(XC)                                                       RE200402
  370 L13=0                                                              RE200403
  371 L13=L13+1                                                          RE200404
      X3=DPULSE+(NPULSE(L13)-1)/REPET(L13)                               RE200405
      WRITE(6,78)NRUN(L13),X3,XDPULS,NPULSE(L13),REPET(L13)              RE200406
   78 FORMAT(1H0,5X,5HNRUN=,I3,2X,13HTRAIN LENGTH=,E10.3,3HSEC,2X,12HPULRE200407
     1SE WIDTH=,E10.3,3HSEC/1H ,5X,17HNUMBER OF PULSES=,I5,3X,16HREPETITRE200408
     2ION RATE=,E10.3,10HPULSES/SEC)                                     RE200409
      IF(IFIL.EQ.0)GO TO 80                                              RE200410
      WRITE(6,79)RIM,LESION                                              RE200411
   79 FORMAT(1H ,5X,12HBEAM RADIUS=,E10.3,2HCM,5X,14HLESION RADIUS=,E10.RE200412
     13,2HCM)                                                            RE200413
      GO TO 82                                                           RE200414
   80 WRITE(6,81)RIM,LESION                                              RE200415
   81 FORMAT(1H ,5X,13HIMAGE RADIUS=,E10.3,2HCM,5X,14HLESION RADIUS=,E10RE200416
     1.3,2HCM)                                                           RE200417
   82 IF(IPRT(8).EQ.0)GO TO 108                                          RE200418
      TC=1./REPET(L13)                                                   RE200419
      NPL=NPULSE(L13)                                                    RE200420
      KX=NP+3                                                            RE200421
      IN=1                                                               RE200422
   83 IF(NPL/IN.LT.20)GO TO 84                                           RE200423
      IN=IN+2                                                            RE200424
      GO TO 83                                                           RE200425
   84 X1=NPL                                                             RE200426
      INX=.5+X1/IN                                                       RE200427
      L1=ALOG(DPULSE)/.69315+29.                                         RE200428
      IF(L1.LT.1)L1=1                                                    RE200429
      INXX=PTIME(L1)*INX                                                 RE200430
C *** STORE TIME INTERVALS AND LOGS OF INTERVALS FOR DAMAGE CALCULATIONSRE200431
      ZT(1)=PTIME/2.                                                     RE200432
      ZTT(1)=ALOG(IN*PTIME)                                              RE200433
      DO 85 L3=2,NP                                                      RE200434
      ZTT(L3)=ALOG(IN*PTIME)                                             RE200435
   85 ZT(L3)=ZT(L3-1)+PTIME                                              RE200436
      L1=NP+1                                                            RE200437
      X3=(TC-DPULSE)/(KX-NP)                                             RE200438
      ZT(L1)=DPULSE+X3/2.                                                RE200439
      ZTT(L1)=ALOG(IN*X3)                                                RE200440
      L1=L1+1                                                            RE200441
      DO 86 L3=L1,KX                                                     RE200442
      ZTT(L3)=ALOG(IN*X3)                                                RE200443
   86 ZT(L3)=ZT(L3-1)+X3                                                 RE200444
C *** CALCULATE TEMPERATURE RISES ASSOCIATED WITH L3-TH TIME INTERVAL    RE200445
C *** FOLLOWING (L6-.5)*IN-.5 PULSE                                      RE200446
      DO 95 L=1,LIJ                                                      RE200447
      DO 95 L3=1,KX                                                      RE200448
      X1=0.                                                              RE200449
      L1=1+IN/2                                                          RE200450
      L7=1                                                               RE200451
   87 X3=(L7-1)*TC+ZT(L3)                                                RE200452
      K=ALOG(X3*X60+1.)/X61+1.                                           RE200453
      X5=VE(L,K,1)+(X3-XT(K))*(VE(L,K+1,1)-VE(L,K,1))/(XT(K+1)-XT(K))    RE200454
      X1=X1+X5                                                           RE200455
      IF(X5.LT..0001*X1)GO TO 88                                         RE200456
```

```
      L7=L7+1
      IF(L7.LE.L1)GO TO 87
   88 VZ(L,1,L3,1)=X1
      DO 93 L6=2,INXX
      IF(X5.LT..0001*X1)GO TO 93
      X1=VZ(L,L6-1,L3,1)
      L2=L1+1
      L1=L1+IN
      L7=L2
   90 X3=(L7-1)*TC+ZT(L3)
      K=ALOG(X3*X60+1.)/X61+1.
      X5=VE(L,K,1)+(X3-XT(K))*(VE(L,K+1,1)-VE(L,K,1))/(XT(K+1)-XT(K))
      X1=X1+X5
      IF(X5.LT..0001*X1)GO TO 93
      L7=L7+1
      IF(L7.LE.L1)GO TO 90
   93 VZ(L,L6,L3,1)=X1
      L1=INX+1
      DO 94 L6=L1,INXX
      L8=L6-INX
   94 VZ(L,L6,L3,1)=VZ(L,L6,L3,1)-VZ(L,L8,L3,1)
   95 CONTINUE
C *** DAMAGE CALCULATIONS --------------
      WRITE(6,375)
  375 FORMAT(1H0,31HPREDICTED THRESHOLD LASER POWER)
      DO 104 L=1,LIJ
      I=ID(L)
      J=JD(L)
      IF(VZ(L,INX,NP,1).LT..001)QD(I,J)=1.E+20
      IF(VZ(L,INX,NP,1).LT..001)GO TO 104
      L9=10.*(.4+EXP(-.0014*DPULSE))/VZ(L,INX,NP,1)
      CQ=L9+1.
      X10=70.*(.4+EXP(-.0014*DPULSE))/VZ(L,INX,NP,1)
      IF(L9.EQ.0)CQ=X10
      LLT=0
      LGT=0
   99 DAMC=0.
      L6=1
  100 DO 101 L3=1,KX
      X3=0.
      X50=VZ(L,L6,L3,1)*CQ+273.+T0
      IF(X50.LT.317.)GO TO 101
      X1=ZTT(L3)+DAMAGE(1,1)-DAMAGE(1,2)/X50
      IF(X50.GT.323.)X1=ZTT(L3)+DAMAGE(2,1)-DAMAGE(2,2)/X50
      IF(X1.GT.0.)X3=1.01
      IF(X1.GT.0.)GO TO 101
      X3=EXP(X1)
  101 DAMC=DAMC+X3
      IF(DAMC.GT.1.)GO TO 102
C *** INCREASE TIME INDICES AND CONTINUE
      L6=L6+1
      IF(L6.LE.INXX)GO TO 100
C *** ADJUST LASER POWER TO YIELD THRESHOLD DAMAGE AT GIVEN POINT
      IF(LGT.EQ.1)CQ=1.02*CQ
      IF(LGT.EQ.1)GO TO 103
      LLT=1
      CQ=1.04*CQ
```

```
      GO TO 99
  102 IF(LLT.EQ.1)CQ=.98*CQ
      IF(LLT.EQ.1)GO TO 103
      LGT=1
      CQ=.96*CQ
      GO TO 99
  103 QD(I,J)=CQ*POX
  104 CONTINUE
      WRITE(6,63) (R(J),J=1,JM)
      DO 97 I=ID1,ID2
      DO 97 J=1,JM
   97 XQD(I,J)=QD(I,J)*XXQ
      DO 106 I=ID1,ID2
      X1=Z(I)-Z(IPE)+DZ/2.
      IF(JM.GT.9)GO TO 385
      WRITE(6,105)X1,(XQD(I,J),J=1,JM)
      GO TO 106
  385 WRITE(6,105)X1,(XQD(I,J),J=1,9)
      WRITE(6,105)X1,(XQD(I,J),J=10,JM)
  105 FORMAT(1H ,2X,2HZ=,F7.5,1X,3HQD=,1P9E13.6)
  106 CONTINUE
  108 IF(KTYPE.EQ.0)GO TO 174
C *** CALCULATE AND STORE (MULTIPLE PULSE EXPOSURE) TEMPERATURES FOR
C *** PLOTTING PROFILES
      TC=1./REPET(L13)
      NPL=NPULSE(L13)
      WRITE(6,139)
      DO 123 L15=1,KTYPE
      IF(TIMEX(L15).GT.XT(KT))GO TO 123
      RGV=0.
      L2=TIMEX(L15)/TC
      DTIME=TIMEX(L15)-L2*TC
      L2=L2+1
      DO 116 I=II1,II2
      DO 116 J=JJ1,JJ2
      X1=0.
      DO 113 L6=1,L2
      K=ALOG((DTIME+(L6-1)*TC)*X60+1.)/X61+1.
      X2=(DTIME+(L6-1)*TC-XT(K))/(XT(K+1)-XT(K))
  113 X1=X1+VC(I,J,K)+X2*(VC(I,J,K+1)-VC(I,J,K))
      V(I,J)=X1
      L3=L2-NPL
      IF(L3.LE.0)GO TO 115
      X1=0.
      DO 114 L6=1,L3
      K=ALOG((DTIME+(L6-1)*TC)*X60+1.)/X61+1.
      X2=(DTIME+(L6-1)*TC-XT(K))/(XT(K+1)-XT(K))
  114 X1=X1+VC(I,J,K)+X2*(VC(I,J,K+1)-VC(I,J,K))
      V(I,J)=V(I,J)-X1
  115 IF(V(I,J).GT.RGV)RGV=V(I,J)
  116 CONTINUE
      SAVRGV(L15)=RGV
      IF(KTYPEO.EQ.1)GO TO 121
      WRITE(7,117)NRUN(L13),NPULSE(L13),REPET(L13)
  117 FORMAT(2I7,E10.4)
      WRITE(7,118)XDPULS,WAVEL,RIM
  118 FORMAT(7E11.4)
```

```
      WRITE(7,119)II1,II2,II3,JJ1,JJ2
  119 FORMAT(5I7)
      WRITE(7,119)N3,M3
      WRITE(7,120) (R(J),J=1,N3)
  120 FORMAT(10F8.4)
      WRITE(7,120) (Z(I),I=1,M3)
      WRITE(7,118)TIMEX(L15)
  121 WRITE(6,141)TIMEX(L15)
      WRITE(6,63) (R(J),J=JJ1,JJ2)
      JCNT=JJ2-JJ1+1
      IF(JCNT.GT.9)GO TO 390
      GO TO 391
  390  JJCNT=JCNT-9
      JJJ2=JJ2-JJCNT
      JJJ2P1=JJJ2+1
  391  DO 122 I=II1,II2
      X1=Z(I)-Z(IPE)+DZ/2.
      IF(JCNT.GT.9)GO TO 392
      WRITE(6,64)X1,(V(I,J),J=JJ1,JJ2)
      GO TO 393
  392  WRITE(6,64)X1,(V(I,J),J=JJ1,JJJ2)
      WRITE(6,64)X1,(V(I,J),J=JJJ2P1,JJ2)
  393 IF(KTYPEO.EQ.1)GO TO 122
      WRITE(7,137)(V(I,J),J=JJ1,JJ2)
  122 CONTINUE
  123 CONTINUE
      RGV=0.
      DO 395 LL15=1,KTYPE
      IF(SAVRGV(LL15).GT.RGV)RGV=SAVRGV(LL15)
  395 CONTINUE
      WRITE(7,396)
  396 FORMAT(22HMAX RGV CARD(S)  FOLLOW)
      DO 397 LL15=1,KTYPE
  397 WRITE(7,137)RGV
      GO TO 174
  124 FORMAT(1H ,5X,1P9E13.6)
  137 FORMAT(6E13.6)
  139 FORMAT(1H0,35HTEMPERATURE RISES AT SELECTED TIMES)
  141 FORMAT(1H0,5X,5HTIME=,E11.4)
  145 IF(L13.EQ.NTEST)GO TO 300
      GO TO 371
C *** DAMAGE CALCULATIONS FOR SINGLE PULSE
C     -------------------------------------------
  125 WRITE(6,126)NRUN(1),XDPULS,NPULSE(1)
  126 FORMAT(1H0,5X,5HNRUN=,I3,2X,12HPULSE WIDTH=,E10.3,2X,17HNUMBER OF
     1PULSES=,I5)
      IF(IFIL.EQ.0)GO TO 127
      WRITE(6,79)RIM,LESION
      GO TO 128
  127 WRITE(6,81)RIM,LESION
  128 IF(IPRT(8).EQ.0)GO TO 150
      WRITE(6,375)
      XQ=0.
      DO 138 I=ID1,ID2
      DO 138 J=1,JM
      IF(VC(I,J,KM).LT..001)QD(I,J)=1.0E+20
      IF(VC(I,J,KM).LT..001)GO TO 138
```

```
      L9=10.*(.4+EXP(-.0014*DPULSE))/VC(I,J,KM)
      CQ=L9+1.
      X10=70.*(.4+EXP(-.0014*DPULSE))/VC(I,J,KM)
      IF(L9.EQ.0)CQ=X10
      LLT=0
      LGT=0
  131 DAMC=0.
      K=2
  132 X13=ALOG(XT(K)-XT(K-1))
      VPX=(VC(I,J,K)+VC(I,J,K-1))/2.
      X3=0.
      X50=VPX*CQ+273.+TO
      IF(X50.LT.317.)GO TO 134
      X1=X13+DAMAGE(1,1)-DAMAGE(1,2)/X50
      IF(X50.GT.323.)X1=X13+DAMAGE(2,1)-DAMAGE(2,2)/X50
      IF(X1.GT.0.)X3=1.01
      IF(X1.GT.0.)GO TO 134
      X3=EXP(X1)
  134 DAMC=DAMC+X3
      IF(DAMC.GE.1.)GO TO 135
      K=K+1
      IF(K.LT.KT)GO TO 132
C *** ADJUST LASER POWER TO YIELD THRESHOLD DAMAGE AT GIVEN POINT
      IF(LGT.EQ.1)CQ=1.02*CQ
      IF(LGT.EQ.1)GO TO 136
      LLT=1
      CQ=1.04*CQ
      GO TO 131
  135 IF(LLT.EQ.1)CQ=.98*CQ
      IF(LLT.EQ.1)GO TO 136
      LGT=1
      CQ=.96*CQ
      GO TO 131
  136 QD(I,J)=CQ*POX
  138 CONTINUE
      WRITE(6,63)(R(J),J=1,JM)
      DO 140 I=ID1,ID2
      DO 140 J=1,JM
  140 XQD(I,J)=QD(I,J)*XXQ
      DO 143 I=ID1,ID2
      X1=Z(I)-Z(IPE)+DZ/2.
      IF(JM.GT.9)GO TO 142
      WRITE(6,105)X1,(XQD(I,J),J=1,JM)
      GO TO 143
  142 WRITE(6,105)X1,(XQD(I,J),J=1,9)
      WRITE(6,105)X1,(XQD(I,J),J=10,JM)
  143 CONTINUE
  150 IF(KTYPE.EQ.0)GO TO 174
C *** CALCULATE AND STORE (SINGLE PULSE EXPOSURE) TEMPERATURES FOR
C *** PLOTTING PROFILES
      WRITE(6,139)
      DO 170 L15=1,KTYPE
      RGV=0.
      DTIME=TIMEX(L15)
      K=ALOG(DTIME*(XC-1.)/DTX+1.)/ALOG(XC)+1.
      IF(K+1.GT.KT)GO TO 170
      X1=(DTIME-XT(K))/(XT(K+1)-XT(K))
```

```
      DO 166 I=II1,II2                                                  RE200685
      DO 166 J=JJ1,JJ2                                                  RE200686
      V(I,J)=VC(I,J,K)+X1*(VC(I,J,K+1)-VC(I,J,K))                      RE200687
      IF(V(I,J).GT.RGV)RGV=V(I,J)                                       RE200688
  166 CONTINUE                                                          RE200689
      SAVRGV(L15)=RGV                                                   RE200690
      IF(KTYPEO.EQ.1)GO TO 167                                          RE200691
      WRITE(7,117)NRUN(1),NPULSE(1),REPET(1)                            RE200692
      WRITE(7,118)XDPULS,WAVEL,RIM                                      RE200693
      WRITE(7,119)II1,II2,II3,JJ1,JJ2                                   RE200694
      WRITE(7,119)N3,M3                                                 RE200695
      WRITE(7,120)(R(J),J=1,N3)                                         RE200696
      WRITE(7,120)(Z(I),I=1,M3)                                         RE200697
      WRITE(7,118)TIMEX(L15)                                            RE200698
  167 WRITE(6,141)TIMEX(L15)                                            RE200699
      WRITE(6,63)(R(J),J=JJ1,JJ2)                                       RE200700
      JCNT=JJ2-JJ1+1                                                    RE200701
      IF(JCNT.GT.9)GO TO 400                                            RE200702
      GO TO 401                                                         RE200703
  400 JJCNT=JCNT-9                                                      RE200704
      JJJ2=JJ2-JJCNT                                                    RE200705
      JJJ2P1=JJJ2+1                                                     RE200706
  401 DO 168 I=II1,II2                                                  RE200707
      X1=Z(I)-Z(IPE)+DZ/2.                                              RE200708
      IF(JCNT.GT.9)GO TO 402                                            RE200709
      WRITE(6,64)X1,(V(I,J),J=JJ1,JJ2)                                  RE200710
      GO TO 403                                                         RE200711
  402 WRITE(6,64)X1,(V(I,J),J=JJ1,JJJ2)                                 RE200712
      WRITE(6,64)X1,(V(I,J),J=JJJ2P1,JJ2)                               RE200713
  403 IF(KTYPEO.EQ.1)GO TO 168                                          RE200714
      WRITE(7,137)(V(I,J),J=JJ1,JJ2)                                    RE200715
  168 CONTINUE                                                          RE200716
  170 CONTINUE                                                          RE200717
      RGV=0.                                                            RE200718
      DO 405 LL15=1,KTYPE                                               RE200719
      IF(SAVRGV(LL15).GT.RGV)RGV=SAVRGV(LL15)                           RE200720
  405 CONTINUE                                                          RE200721
      WRITE(7,396)                                                      RE200722
      DO 406 LL15=1,KTYPE                                               RE200723
  406 WRITE(7,137)RGV                                                   RE200724
C *** INTERPOLATE AXIAL EXTENT OF DAMAGE                                RE200725
  174 I5=0                                                              RE200726
      I6=0                                                              RE200727
      IF(ID1.EQ.ID2)GO TO 182                                           RE200728
      DO 175 I=ID1,ID2                                                  RE200729
      L1=ID1+ID2-I                                                      RE200730
      IF(QD(L1,1).GT.POX)I5=L1                                          RE200731
      IF(QD(L1,1).LT.POX)I6=L1                                          RE200732
      IF(QD(I,1).GT.POX)I7=I                                            RE200733
      IF(QD(I,1).LT.POX)I8=I                                            RE200734
  175 CONTINUE                                                          RE200735
      IF(IPRT(9).EQ.0)GO TO 182                                         RE200736
      WRITE(6,350)                                                      RE200737
  350 FORMAT(1H0,22HAXIAL EXTENT OF DAMAGE)                             RE200738
      IF(I5.EQ.0)WRITE(6,176)                                           RE200739
  176 FORMAT(1H0,5X,45HDEPTHS OF DAMAGE BEYOND BOTH SPECIFIED DEPTHS)   RE200740
      IF(I5.EQ.0)GO TO 182                                              RE200741
```

```
      IF(I6.EQ.0)GO TO 190
      IF(I5.GE.I6)GO TO 178
      X2=ALOG(QD(I6,1)/QD(I5,1))/(Z(I6)-Z(I5))
      X1=QD(I5,1)
      X3=ALOG(POX/X1)/X2+Z(I5)-Z(IPE)+DZ/2.
      WRITE(6,177)X3
  177 FORMAT(1H0,5X,24HMINIMUM DEPTH OF DAMAGE=,E10.3,2HCM)
  178 IF(I8.GE.I7)GO TO 182
      X2=ALOG(QD(I8,1)/QD(I7,1))/(Z(I8)-Z(I7))
      X1=QD(I7,1)
      X3=ALOG(POX/X1)/X2+Z(I7)-Z(IPE)+DZ/2.
  180 WRITE(6,181)X3
  181 FORMAT(1H0,5X,24HMAXIMUM DEPTH OF DAMAGE=,E10.3,2HCM)
C *** INTERPOLATE RADIAL EXTENT OF IRREVERSIBLE DAMAGE AT SPECIFIED
C *** DEPTHS
  182 IF(IPRT(10).EQ.0)GO TO 192
      WRITE(6,360)
  360 FORMAT(1H0,23HRADIAL EXTENT OF DAMAGE)
      DO 189 I=ID1,ID2
      J1=0
      X3=Z(I)-Z(IPE)+DZ/2.
      DO 183 J=1,JM
      IF(POX.GT.QD(I,J))J1=J
  183 CONTINUE
      X20=0.
      IF(J1.EQ.0)GO TO 187
      IF(J1.EQ.JM)WRITE(6,185)X3,R(JM)
  185 FORMAT(1H0,5X,2HZ=,E9.3,2HCM,5X,36HRADIAL EXTENT OF DAMAGE GREATER
     1 THAN,E10.3,2HCM)
      IF(J1.EQ.JM)GO TO 189
      X2=ALOG(QD(I,J1+1)/QD(I,J1))/(R(J1+1)-R(J1))
      X1=QD(I,J1)
      X20=ALOG(POX/X1)/X2+R(J1)
  187 WRITE(6,188)X3,X20
  188 FORMAT(1H0,5X,2HZ=,E9.3,2HCM,5X,37HRADIAL EXTENT OF IRREVERSIBLE D
     1AMAGE=,E10.3,2HCM)
  189 CONTINUE
      IF(LPX.EQ.0)GO TO 300
      GO TO 145
  190 WRITE(6,191)
  191 FORMAT(1H0,5X,31HNO DAMAGE---LASER POWER TOO LOW)
  192 IF(LPX.EQ.0)GO TO 300
      GO TO 145
  200 STOP
      END
      SUBROUTINE GRID
C *** GRID COMPUTES THE COEFFICIENTS IN PARTIAL DIFFERENTIAL EQUATIONS A
C *** RADIAL AND AXIAL COORDINATES, R AND Z, AND ASSIGNS CONDUCTIVITY AN
C *** VOLUMETRIC SPECIFIC HEAT TO GRID
C *** CALCULATE B(CM**-2) AND R (CM)
      COMMON A(29,3),AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),
     2IAB(29,14),IBLOOD(10),IFIL,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,K,KM,KT,M,M1,M2,
     4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TSC,TTS,V(29,14)
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),
```

```
     7XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC
      DIMENSION IX(7),LX(7)
C *** CALCULATE B(CM**-2) AND R (CM)
      WRITE(6,170)
  170 FORMAT(1H1)
      R(1)=0.
      CK=N-N1
      CP=RVL/DR-N1+1.
      X1=2.
  180 R2=EXP(ALOG(2.*(CP*(X1-1.)+1.)/(X1+1.))/(CK-1.))
      IF(R2/X1.GT..99999.AND.R2/X1.LT.1.00001)GO TO 181
      X1=R2
      GO TO 180
  181 IF(IPRT(1).EQ.0)GO TO 220
      WRITE(6,182)
  182 FORMAT(1H0,16HGRID INFORMATION)
      WRITE(6,184)R2
  184 FORMAT(1H0,5X,3HR2=,F8.4)
  220 RN=DR*(N1-1.+(R2**(CK+1.)-1.)/(R2-1.))
C *** CALCULATE RADIAL SPACE STEPS R(J)
      DO 185 J=2,N4
  185 R(J)=DR*(J-1)
      X1=R2*DR
      DO 186 J=N4,N
      R(J+1)=R(J)+X1
  186 X1=R2*X1
C *** CALCULATE COEFFICIENTS B OF FINITE DIFFERENCE EQNS.
      X1=2./(DR*DR)
      DO 187 J=2,N1
      B(J,1)=.25*(2*J-3)*X1/(J-1)
      B(J,2)=X1
  187 B(J,3)=X1-B(J,1)
      X2=DR
      X1=R2*DR
      DO 188 J=N4,N
      B(J,2)=2./(X1*X2)
      B(J,1)=(2./X2-1./R(J))/(X1+X2)
      B(J,3)=B(J,2)-B(J,1)
      X2=R2*X2
  188 X1=R2*X1
      B(1,1)=0.
      B(1,2)=2./(DR*DR)
      B(1,3)=B(1,2)
      DO 189 J=1,N
      IF(R(J).LT.RVL)JVL=J
  189 CONTINUE
C *** CALCULATE AXIAL SPACE STEPS Z(I)
      CK=M2-M1+1
      X1=2.
  190 CP=2.*TAV/DZ+1.-(X1**(CK-1.)-1.)/(X1-1.)
      R1=EXP(ALOG(CP*X1-CP+1.)/CK)
      IF(R1/X1.GT..99999.AND.R1/X1.LT.1.00001)GO TO 192
      X1=R1
      GO TO 190
  192 ZM=((R1**CK-1.)/(R1-1.)+M1-1.)*DZ
      IF(IPRT(1).EQ.0)GO TO 230
```

```
      WRITE(6,194)R1,ZM
  194 FORMAT(1H ,5X,3HR1=,F8.4,2X,3HZM=,F8.4)
  230 X1=DZ
      X2=X1
      DO 195 I=2,M2
      Z(M2+I)=ZM+X2
      Z(M2+2-I)=ZM-X2
      IF(I.GT.M1) X1=R1*X1
  195 X2=X2+X1
      Z(1)=0.
      Z(M2+1)=ZM
      Z(M+1)=2.*ZM
      X1=Z(IPE)-DZ/2.-ZD(2)
      DO 196 I=1,M3
  196 Z(I)=Z(I)-X1
      L3=IPA
      DO 200 L=1,7
      L1=0
      DO 197 I=IPA,M3
      IF(Z(I).LT.ZD(L+1))L3=I
      IF(Z(I).LT.ZD(L).OR.Z(I).GE.ZD(L+1))GO TO 197
      L2=I
      L1=L1+1
  197 CONTINUE
      IF(L1.EQ.0)IX(L)=L3
      IF(L1.EQ.0)LX(L)=L3
      IF(L1.GT.0)IX(L)=L2+1-L1
      IF(L1.GT.0)LX(L)=L2
  200 CONTINUE
      IPV=IX(4)
      IPC=IX(5)
      IPS=IX(6)
      IPT=IX(7)
      LPA=LX(1)
      LPE=LX(3)
      LPV=LX(4)
      LPC=LX(5)
      LPS=LX(6)
      LPT=M3
C *** SET CONDUCTIVITY CON AND HEAT CAPACITY VSH FOR VARIOUS EYE MEDIA
      DO 203 I=1,LPA
      CON(I)=CONX(1)
  203 VSH(I)=VSHX(1)
      DO 204 I=IPE,LPE
      CON(I)=CONX(2)
  204 VSH(I)=VSHX(2)
      DO 205 I=IPV,LPV
      CON(I)=CONX(3)
  205 VSH(I)=VSHX(3)
      DO 206 I=IPC,LPC
      CON(I)=CONX(4)
  206 VSH(I)=VSHX(4)
      DO 207 I=IPS,LPS
      CON(I)=CONX(5)
  207 VSH(I)=VSHX(5)
      DO 208 I=IPT,M3
      CON(I)=CONX(6)
```

```
  208 VSH(I)=VSHX(6)
C *** CALCULATE COEFFICIENTS A OF FINITE DIFFERENCE EQNS.
      DO 210 I=IPA,M
      X1=Z(I+1)-Z(I-1)
      X2=(CON(I-1)-CON(I+1))/(X1*X1)
      X3=2.*CON(I)/X1
      A(I,1)=X2+X3/(Z(I)-Z(I-1))
      IF(I.EQ.IPA)A(I,1)=0.
      A(I,3)=-X2+X3/(Z(I+1)-Z(I))
  210 A(I,2)=A(I,1)+A(I,3)
      RETURN
      END
      SUBROUTINE IMAGE
C *** IMAGE COMPUTES THE RETINAL IRRADIANCE PROFILE
      COMMON A(29,3),AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),
     2IAB(29,14),IBLOOD(10),IFIL,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,K,KM,KT,M,M1,M2,
     4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TSC,TTS,V(29,14)
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),
     7XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC
      DIMENSION FA(2001),FP(2001),FX(2001),FY(2001),JO(32),NA(22),PX(30)
     1,RX(30),XF1(2001),XF2(2001)
      REAL JO,NA,NB,NC
      DO 200 J=1,N
  200 PR(J)=0.
      LI=500
      LII=LI
      DO 201 L=1,LI
  201 FX(L)=0.
      READ(5,202)PUPIL
  202 FORMAT(10E8.3)
      RINT=PUPIL/(LI-1)
      IF(IPROF.EQ.1)GO TO 214
      IF(IPROF.EQ.0)GO TO 219
C *** INTERPOLATE IRREGULAR LASER PROFILE(SYMMETRIC IN R) AT INTERVALS
C *** OF RINT STARTING AT R=0
      READ(5,205)LR
  205 FORMAT(I7)
      READ(5,206)(RX(L),L=1,LR)
  206 FORMAT(10E7.3)
      READ(5,206)(PX(L),L=1,LR)
      X1=PX(1)
      DO 207 L=1,LR
  207 PX(L)=PX(L)/X1
      X5=0.
      X6=0.
      DO 208 L=2,LR
      X2=(PX(L)-PX(L-1))/(RX(L)-RX(L-1))
      X1=PX(L-1)-X2*RX(L-1)
      X3=X1*(RX(L)*RX(L)-RX(L-1)*RX(L-1))/2.
      X4=X2*(RX(L)*RX(L)*RX(L)-RX(L-1)*RX(L-1)*RX(L-1))/3.
      IF(RX(L).GT.PUPIL)X6=X6+6.2832*(X3+X4)
  208 X5=X5+6.2832*(X3+X4)
      QP=POX*.23906*(1.-RCO)/X5
```

```
      XX=(X5-X6)/X5
      IF(RX(LR).LT.PUPIL)LII=RX(LR)/RINT+1
      L2=2
      X1=0.
      DO 213 L=1,LII
  210 IF(RX(L2).GT.X1)GO TO 212
      L2=L2+1
      IF(L2.LE.LR)GO TO 210
      GO TO 213
  212 X2=(X1-RX(L2-1))/(RX(L2)-RX(L2-1))
      FX(L)=PX(L2-1)+X2*(PX(L2)-PX(L2-1))
  213 X1=X1+RINT
      GO TO 223
C *** CALCULATE GAUSSIAN LASER PROFILE AT INTERVALS OF RINT STARTING AT
  214 SIGMA=RIM*SQRT(-2./ALOG(CUT))
      QP=2.*POX*.23906*(1.-RCO)/(3.1416*SIGMA*SIGMA)
      XX=1.-EXP(-2.*PUPIL*PUPIL/(SIGMA*SIGMA))
      IF(IFIL.EQ.1)GO TO 217
      DO 216 J=1,N
      X3=2.*R(J)*R(J)/(SIGMA*SIGMA)
      IF(X3.GT.80.)GO TO 216
      PR(J)=EXP(-X3)
  216 CONTINUE
      GO TO 276
  217 X1=0.
      DO 218 L=1,LII
      X3=2.*X1*X1/(SIGMA*SIGMA)
      FX(L)=0.
      IF(X3.GT.80.)GO TO 218
      FX(L)=EXP(-X3)
  218 X1=X1+RINT
      GO TO 227
C *** SPECIFY UNIFORM LASER PROFILE FROM R(1) TO R(LIM)
  219 QP=POX*.23906*(1.-RCO)/(3.1416*RIM*RIM)
      XX=1.
      IF(RIM.GT.PUPIL)XX=PUPIL*PUPIL/(RIM*RIM)
      IF(IFIL.EQ.1)GO TO 221
      DO 220 J=1,LIM
  220 PR(J)=1.
      GO TO 276
  221 L1=RIM/RINT
      RINT=RIM/L1
      LII=RIM/RINT+1
      DO 222 L=1,LII
  222 FX(L)=1.
      GO TO 227
C *** CALCULATE TOTAL AREA FA(L) AND PORTION OF LASERS POWER BETWEEN R=0
C *** AND (L-.5)*RINT
  223 IF(IFIL.EQ.1)GO TO 227
      FP(1)=3.1416*FX(1)*RINT*RINT/4.
      FA(1)=3.1416*RINT*RINT/4.
      DO 224 L=2,LII
      X1=(L-.5)*RINT
      X2=(L-1.5)*RINT
      FP(L)=FP(L-1)+FX(L)*3.1416*(X1*X1-X2*X2)
  224 FA(L)=FA(L-1)+3.1416*(X1*X1-X2*X2)
C *** CALCULATE PROFILE PR(J)
```

```
      X1=0.
      X2=0.
      DO 225 J=1,N
      X3=(R(J)+R(J+1))/(2.*RINT)+.5000001
      IF(X3.LT.1.)X3=1.000001
      L2=X3
      IF(L2.GE.LII)GO TO 225
      X4=X3-L2
      X5=FP(L2)+X4*(FP(L2+1)-FP(L2))
      X6=FA(L2)+X4*(FA(L2+1)-FA(L2))
      PR(J)=(X5-X1)/(X6-X2)
      X1=X5
      X2=X6
  225 CONTINUE
      GO TO 276
C *** SPREAD FUNCTION CALCULATIONS
  227 READ(5,202)ZO,FLO,FC,NB,CABER,PP,PC
      CABER2=CABER/WAVEL
      READ(5,228)(JO(L),L=1,32)
  228 FORMAT(10F8.5)
      READ(5,228)(NA(L),L=1,22)
      X1=(WAVEL-350.)/50.+1.
      L1=X1
      X2=X1-L1
      NC=NA(L1)+X2*(NA(L1+1)-NA(L1))
      X1=(NB-1.)*NC/(NB*(NC-1.))
      FL=FLO*X1
      X2=ZO/FLO
      X0=NC*ZO*X1/(NC*X2-X1)-FLO
      X3=1.-PC*(NC*ZO-FC)/(NC*ZO*FC)
      DO 230 L=1,LI
      IF(L.GT.LII)GO TO 230
      X1=(L-1)/X3+1.000001
      L1=X1
      X2=X1-L1
      IF(L1+1.GT.LI)FY(L)=0.
      IF(L1+1.GT.LI)LII=L
      IF(L1+1.GT.LI)GO TO 230
      FY(L)=(FX(L1)+X2*(FX(L1+1)-FX(L1)))/(X3*X3)
  230 CONTINUE
      DO 231 L=1,LII
  231 FX(L)=FY(L)
      X5=ATAN(PUPIL/(FLO-PP+X0))
      X6=1.-COS(X5)
      X7=SIN(X5)*SIN(X5)
      FF=FLO-PP
      DO 234 L=1,LII
      X4=(L-1)*RINT
      X1=6.2832*NC*(-FF-X6*X0+SQRT(FF*FF-X7*X0*X0))*X4*X4/(WAVEL*1.E-7*
     1PUPIL*PUPIL)
      X2=CABER2*X4*X4*X4*X4
      XF1(L)=SQRT(FX(L))*COS(X1+X2)
  234 XF2(L)=SQRT(FX(L))*SIN(X1+X2)
      DO 260 J=1,N
      X1=6.2832*R(J)/(WAVEL*1.E-7*FF)
      X2=0.
      X3=0.
```

```
      DO 255 L=1,LII
      X4=X1*(L-1)*RINT
      IF(L.EQ.1)X4=X1*.25*RINT
      IF(X4.GT.3.)GO TO 250
      X5=X4/.1+1.000001
      L1=X5
      X5=X5-L1
      X7=JO(L1)+X5*(JO(L1+1)-JO(L1))
      GO TO 251
  250 X6=3./X4
      X8=.79788456-.00000077*X6-.00552740*X6*X6-.00009512*X6*X6*X6+
     1.00137237*X6*X6*X6*X6-.00072805*X6*X6*X6*X6*X6+.00014476*X6*X6*X6*
     2X6*X6*X6
      X9=X4-.78539816-.04166397*X6-.00003954*X6*X6+.00262573*X6*X6*X6-
     1.00054125*X6*X6*X6*X6-.00029333*X6*X6*X6*X6*X6+.00013558*X6*X6*X6*
     2X6*X6*X6
      X7=X8*COS(X9)/SQRT(X4)
  251 IF(L.GT.1)GO TO 252
      X2=X2+X7*.25*(3.*XF1(1)+XF1(2))*.25*RINT*.5*RINT
      X3=X3+X7*.25*(3.*XF2(1)+XF2(2))*.25*RINT*.5*RINT
      GO TO 255
  252 X2=X2+X7*XF1(L)*(L-1)*RINT*RINT
      X3=X3+X7*XF2(L)*(L-1)*RINT*RINT
  255 CONTINUE
  260 HR(J)=X2*X2+X3*X3
      X1=HR(1)
      DO 270 J=1,N
  270 HR(J)=HR(J)/X1
      X1=.0002
      X2=3.1416*X1*X1/4
      J=2
      X4=HR(1)*X2
      L1=2
  271 IF(X1.LT.R(J)+.0000001)GO TO 272
      J=J+1
      GO TO 271
  272 X5=(X1-R(J-1))/(R(J)-R(J-1))
      X6=HR(J-1)+X5*(HR(J)-HR(J-1))
      X7=8.*(L1-1)*X2
      X4=X4+X6*X7
      L1=L1+1
      X1=X1+.0002
      IF(X1.LE..1)GO TO 271
      QP=.23906*XX*POX*(1.-RCO)/X4
      RETURN
  276 DO 280 J=1,N
  280 HR(J)=PR(J)
      RETURN
      END
      SUBROUTINE HTXDEP
C *** HTXDEP COMPUTES RATE OF HEAT DEPOSITON AT VARIOUS POINTS I,J
      COMMON A(29,3),AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),
     2IAB(29,14),IBLOOD(10),IFIL,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,K,KM,KT,M,M1,M2,
     4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TSC,TTS,V(29,14)
```

```
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),RE201141
     7XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,           RE201142
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC        RE201143
      DIMENSION AB(29,3),ABR(29,7),ABS(7),II(29),IZ(29),REF(8),REFL(8), RE201144
     1ZH(29)                                                            RE201145
      IF(IHT.EQ.0)RETURN                                                RE201146
      IF(QP.LT.1.E-25)GO TO 340                                         RE201147
      IF(IHT.EQ.1)RETURN                                                RE201148
      LZ=7                                                              RE201149
      LZ0=LZ-1                                                          RE201150
      LZ1=LZ+1                                                          RE201151
      DO 280 I=1,M                                                      RE201152
      II(I)=0                                                           RE201153
      IZ(I)=0                                                           RE201154
      ZH(I)=(Z(I)+Z(I+1))/2.                                            RE201155
      DO 279 L1=1,3                                                     RE201156
  279 AB(I,L1)=0.                                                       RE201157
      DO 280 L1=1,LZ                                                    RE201158
  280 ABR(I,L1)=0.                                                      RE201159
      DO 282 L1=1,LZ                                                    RE201160
      REF(L1)=0.                                                        RE201161
  282 REFL(L1)=0.                                                       RE201162
      REF(2)=RRT                                                        RE201163
      REF(6)=RSC                                                        RE201164
      REF(LZ1)=0.                                                       RE201165
      IF(IPRT(1).EQ.0)GO TO 350                                         RE201166
      WRITE(6,283)(ZH(I),I=1,M)                                         RE201167
  283 FORMAT(1H0,5X,3HZH=/(1H ,5X,10E10.3))                             RE201168
C *** EVALUATE ABSORPTION CONSTANTS APE1 AND APE2 FOR FRONT AND REAR OF RE201169
C *** PE                                                                RE201170
  350 IF(IGX.EQ.1)GO TO 284                                             RE201171
      APE1=(APE-ACH*(1.-RPE))/RPE                                       RE201172
      APE2=ACH                                                          RE201173
      GO TO 285                                                         RE201174
  284 APE1=ACH                                                          RE201175
      APE2=(APE-ACH*RPE)/(1.-RPE)                                       RE201176
  285 ABS(1)=AAV                                                        RE201177
      ABS(2)=APE1                                                       RE201178
      ABS(3)=APE2                                                       RE201179
      ABS(4)=AVL                                                        RE201180
      ABS(5)=ACH                                                        RE201181
      ABS(6)=ASC                                                        RE201182
      ABS(7)=ATS                                                        RE201183
      L1=2                                                              RE201184
      DO 306 I=IPA,M                                                    RE201185
  295 IF(ZH(I-1).LT.ZD(L1))GO TO 296                                    RE201186
      L1=L1+1                                                           RE201187
      GO TO 295                                                         RE201188
  296 IF(ZH(I).GE.ZD(L1))GO TO 299                                      RE201189
C *** NO ZD BETWEEN ZH(I-1) AND ZH(I)                                   RE201190
      AB(I,1)=ABS(L1-1)*(ZH(I)-ZH(I-1))                                 RE201191
      II(I)=1                                                           RE201192
      IZ(I)=L1                                                          RE201193
      IF(L1.GT.LZ)GO TO 306                                             RE201194
      DO 297 L2=L1,LZ                                                   RE201195
  297 ABR(I,L2)=AB(I,1)                                                 RE201196
      GO TO 306                                                         RE201197
```

```
  299 IF(ZH(I).GE.ZD(L1+1))GO TO 303
C *** ONLY ZD(L1) BETWEEN ZH(I-1) AND ZH(I)
      AB(I,1)=ABS(L1-1)*(ZD(L1)-ZH(I-1))
      AB(I,2)=ABS(L1)*(ZH(I)-ZD(L1))
      ABR(I,L1)=AB(I,1)
      II(I)=2
      IZ(I)=L1
      L3=L1+1
      IF(L3.GT.LZ)GO TO 306
      DO 300 L2=L3,LZ
  300 ABR(I,L2)=AB(I,1)+AB(I,2)
      GO TO 306
C *** ZD(L1) AND ZD(L1+1) BETWEEN ZH(I-1) AND ZH(I)
  303 AB(I,1)=ABS(L1-1)*(ZD(L1)-ZH(I-1))
      AB(I,2)=ABS(L1)*(ZD(L1+1)-ZD(L1))
      AB(I,3)=ABS(L1+1)*(ZH(I)-ZD(L1+1))
      ABR(I,L1)=AB(I,1)
      ABR(I,L1+1)=AB(I,1)+AB(I,2)
      II(I)=3
      IZ(I)=L1
      L3=L1+2
      IF(L3.GT.LZ)GO TO 306
      DO 304 L2=L3,LZ
  304 ABR(I,L2)=AB(I,1)+AB(I,2)+AB(I,3)
  306 CONTINUE
      DO 314 I=IPA,M
      IF(AB(I,1).GT.10.)AB(I,1)=10.
      IF(AB(I,2).GT.10.)AB(I,2)=10.
      IF(AB(I,3).GT.10.)AB(I,3)=10.
      DO 314 L=2,LZ
      IF(ABR(I,L).GT.10.)ABR(I,L)=10.
  314 CONTINUE
C *** DEPOSITION BY INCOMING BEAM
      X2=QP
      L1=2
      DO 317 I=IPA,M
      L2=II(I)
      X3=X2
      X2=X2*EXP(-AB(I,1))
      X4=0.
      IF(L2.EQ.1)GO TO 315
      L3=IZ(I)
      X4=X2*REF(L3)
      X2=X2*(1.-REF(L3))*EXP(-AB(I,2))
      IF(L2.EQ.2)GO TO 315
      X4=X4+X2*REF(L3+1)
      X2=X2*(1.-REF(L3+1))*EXP(-AB(I,3))
  315 IF(X2.LT.1.E-10)X2=0.
      DO 317 J=1,JVL
      S(I,J)=(X3-X2-X4)*HR(J)/(ZH(I)-ZH(I-1))
      IF(S(I,J).LT.1.E-10/DPULSE)S(I,J)=0.
  317 CONTINUE
C *** CALCULATION OF REFLECTED INTENSITIES BY VARIOUS INTERFACES
C *** STARTING WITH FIRST INTERNAL INTERFACE
      X2=QP
      DO 322 L1=1,LZO
      X3=ABS(L1)*(ZD(L1+1)-ZD(L1))
```

```
      IF(X3.GT.10.)X3=10.
      X2=X2*EXP(-X3)
      REFL(L1+1)=X2*REF(L1+1)
  322 X2=X2*(1.-REF(L1+1))
      DO 327 L1=2,LZ
      I=IPA
  324 IF(ZH(I).GT.ZD(L1))GO TO 325
      I=I+1
      IF(I.LE.M)GO TO 324
      GO TO 327
  325 X2=REFL(L1)
      DO 326 L3=IPA,I
      X3=X2
      L4=I+IPA-L3
      X2=X2*EXP(-ABR(L4,L1))
      DO 326 J=1,JVL
      S(L4,J)=S(L4,J)+(X3-X2)*HR(J)/(ZH(L4)-ZH(L4-1))
      IF(S(L4,J).LT.1.E-10/DPULSE)S(L4,J)=0.
  326 CONTINUE
  327 CONTINUE
      IHT=1
      RETURN
C *** NO HEAT DEPOSITION,BEAM OFF
  340 DO 342 I=1,M3
      DO 342 J=1,N3
  342 S(I,J)=0.
      IHT=0
      RETURN
      END
      SUBROUTINE BLOOD
C     SUBROUTINE BLOOD COMPUTES CHANGES IN MATRIX ELEMENTS A AND B DUE
C     TO BLOOD FLOW
      COMMON A(29,3),AAV,ACH,APE,ASC,ATS,AVL,B(14,3),BB,BV(14,3),
     1CONX(6),CON(29),CUT,DFLOW(6),DPULSE,DR,DT,DTX,DZ,FL,HR(14),
     2IAB(29,14),IBLOOD(10),IFIL,IGX,IHT,IPA,IPC,IPE,IPROF,IPS,IPT,
     3IPV,IV(29),JVL,LIM,LPA,LPC,LPE,LPS,LPV,LPX,K,KM,KT,M,M1,M2,
     4M3,N,N1,N3,N4,NVL,POX,PR(14),PTIME,QP,R(14),RCO,RIM,RN,RPE,RRT,
     5RVL,RSC,S(29,14),SHB,TAV,TCH,TOM,TPE,TVL,TSC,TTS,V(29,14)
     6,VC(29,14,120),VSH(29),VSHX(6),WAVEL,XC,XFLOW,XFLOWI(6),XFLOWO(6),
     7XT(120),Z(29),ZD(8),ZM,FLOWI(14),FLOWX(14),PUPIL,SIGMA,
     8IPRT(10),APE1,APE2,RINT,ZO,FLO,CABER,CABER2,PP,PC,NB,NC,FC
      DIMENSION RD(14),RH(14),XI(14),XO(14)
C *** INITIAL EVALUATION OF PARAMETERS AND ARRAYS
      DO 800 J=1,N3
      BV(J,1)=0.
      BV(J,2)=0.
      BV(J,3)=0.
      FLOWI(J)=0.
  800 FLOWX(J)=0.
      RH(1)=R(2)/2.
      DO 803 J=2,JVL
  803 RH(J)=(R(J)+R(J+1))/2.
      L2=2
      DO 810 J=1,JVL
  805 IF(DFLOW(L2).GT.RH(J))GO TO 806
      L2=L2+1
      GO TO 805
```

```
806 X1=DFLOW(L2)-DFLOW(L2-1)
    X2=RH(J)-DFLOW(L2-1)
    X3=X2/X1
    XI(J)= XFLOWI(L2-1)+X3*(XFLOWI(L2)-XFLOWI(L2-1))
810 XO(J)= XFLOWO(L2-1)+X3*(XFLOWO(L2)-XFLOWO(L2-1))
    FLOWX(1)=0.
    DO 812 J=2,JVL
812 FLOWX(J)=FLOWX(J-1)+(XI(J-1)-XO(J-1))*(R(J)*R(J)-R(J-1)*R(J-1))/
   1(2.*TVL)
    FLOWX(JVL+1)=FLOWX(JVL)
    L2=2
    FLOWI(1)=XFLOWI(1)/TVL
    DO 820 J=2,JVL
814 IF(DFLOW(L2).GT.R(J))GO TO 816
    L2=L2+1
    GO TO 814
816 X4=DFLOW(L2)-DFLOW(L2-1)
    X5=R(J)-DFLOW(L2-1)
    X6=X5/X4
820 FLOWI(J)=(XFLOWI(L2-1)+X6*(XFLOWI(L2)-XFLOWI(L2-1)))/TVL
    DO 823 J=2,JVL
823 RD(J)=1./(R(J)*(R(J+1)-R(J-1)))
C *** CALCULATE CHANGES IN MATRIX ELEMENTS A AND B DUE TO BLOOD FLOW
    BV(1,1)=0.
    BV(1,2)=-SHB*FLOWI(1)/2.
    BV(1,3)=0.
    BB=-SHB*XFLOW/2.
    DO 825 J=2,JVL
    BV(J,1)=SHB*RD(J)*FLOWX(J)
    BV(J,2)=SHB*RD(J)*(FLOWX(J-1)-FLOWX(J+1))/2.-SHB*FLOWI(J)/2.
825 BV(J,3)=-SHB*RD(J)*FLOWX(J)
    DO 835 I=IPA,M
835 IV(I)=0
    DO 840 L3=1,NVL
    L4=IBLOOD(L3)
840 IV(L4)=1
    DO 845 I=IPA,LPS
    DO 842 J=1,JVL
842 IAB(I,J)=0
    IF(JVL.EQ.N)GO TO 845
    L1=JVL+1
    DO 843 J=L1,N
843 IAB(I,J)=1
845 CONTINUE
    DO 850 I=IPT,M
    DO 850 J=1,N
850 IAB(I,J)=1
    RETURN
    END
```

2694 RECORDS PRINTED

```
C                         PLOTTING ROUTINE  IITRI
C                         VERSION  14 NOV 1975
C  TWO AND THREE DIMENSIONAL PLOTS
C
C     II1,II2        I VALUES DESIGNATING RANGE OF Z(I) VALUES FOR
C                    PLOTTING RANGE=Z(II1) TO Z(II2)
C     II3            DESIGNATED PLANE OR SURFACE CURVE MARKED WITH
C                    AN ASTERISK SYMBOL
C     JJ1,JJ2        J VALUES DESIGNATING RANGE OF R(J) VALUES FOR
C                    PLOTTING RANGE=R(JJ1) TO R(JJ2)
C     R(J)           ORDINATE,CM
C     RGR            RANGE OF R VALUES TO BE PLOTTED,CM
C     RGV            RANGE OF TEMPERATURE VALUES TO BE PLOTTED,C
C     RGZ            RANGE OF Z VALUES TO BE PLOTTED,CM
C     TIMEX          TIME AT WHICH TEMPERATURE RISE VALUES ARE PLOTTED,SEC
C     V(I,J)         TEMPERATURE RISE AT TIME TIMEX(K),C
C     Z(I)           ABSCISSA,CM
      REAL LA
      COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX
      COMMON/PLBAS2/AP(16),AV(16),CP(16),DAT(8)
      DIMENSION LA(4)
      DIMENSION RR(100),PT(3),RP(100)
      DIMENSION R(14),V(29,14),Z(29)
      DATA LA/4HZ,CM,4HR,CM,4H T,C,4HRUN=/
      IPLTX=0
    5 DAT(1)=1.0
      IRR=0
      CALL SSPLOT
      READ(5,9,END=50)NRUN,NPULSE,REPET
    9 FORMAT(2I7,E10.4)
      READ(5,10)DPULSE,WAVEL,RIM
   10 FORMAT(3E11.4)
      READ(5,11)II1,II2,II3,JJ1,JJ2
   11 FORMAT(5I7)
      READ(5,11)N3,M3
      READ(5,12)(R(J),J=1,N3)
   12 FORMAT(10F8.4)
      READ(5,12)(Z(I),I=1,M3)
      READ(5,10)TIMEX
      DO 15 I=II1,II2
      READ(5,16)(V(I,J),J=JJ1,JJ2)
   15 CONTINUE
   16 FORMAT(6E13.6)
      READ(5,16)RGV
C *** START OF PROGRAM FOR PLOTTING
      RGR=R(JJ2)-R(JJ1)
      RGZ=Z(II2)-Z(II1)
      SFLAG=0.
      SFAC=0.
      IF(RGV.LT.1.)GO TO 25
      SFLAG=1.
      IF((RGV.GE.12.).AND.(RGV.LT.112.))SFAC=10.
      IF((RGV.GE.112.).AND.(RGV.LT.1120.))SFAC=100.
      IF((RGV.GE.1120.).AND.(RGV.LT.11200.))SFAC=1000.
      IF(RGV.GE.11200.)SFAC=10000.
      IF(SFAC.EQ.0.)GO TO 26
      DO 14 I=II1,II2
```

```
      DO 13 J=JJ1,JJ2
   13 V(I,J)=V(I,J)/SFAC
   14 CONTINUE
      RGV=RGV/SFAC
      GO TO 26
   25 IF((RGV.LT.1.).AND.(RGV.GE..1))SFAC=10.
      IF((RGV.LT..1).AND.(RGV.GE..01))SFAC=100.
      IF((RGV.LT..01).AND.(RGV.GE..001))SFAC=1000.
      IF((RGV.LT..001).AND.(RGV.GE..0001))SFAC=10000.
      IF(RGV.LT..0001)SFAC=100000.
      DO 18 I=II1,II2
      DO 17 J=JJ1,JJ2
   17 V(I,J)=V(I,J)*SFAC
   18 CONTINUE
      RGV=SFAC*RGV
   26 WRITE(6,19)
   19 FORMAT(1H1,3X,21HSCIENTIFIC INPUT DATA)
      WRITE(6,21)RGZ,RGR,RGV
   21 FORMAT(1H0,4HRGZ=,E8.3,2X,4HRGR=,E8.3,2X,4HRGV=,E8.3)
      IF(SFAC.EQ.0.)GO TO 28
      IF(SFLAG.NE.0.)GO TO 8
      WRITE(6,7)SFAC
    7 FORMAT(1H0,30HTEMPERATURE RISES SCALED UP BY,F9.1)
      GO TO 28
    8 WRITE(6,27)SFAC
   27 FORMAT(1H0,32HTEMPERATURE RISES SCALED DOWN BY,F9.1)
   28 DO 23 I=II1,II2
      WRITE(6,22)I,(V(I,J),J=JJ1,JJ2)
   22 FORMAT(1H0,2HI=,I3/(1X,10F10.5))
   23 CONTINUE
      WRITE(6,24)
   24 FORMAT(1H0,3X,35HAXIS INFORMATION (SYSTEM GENERATED)/)
C *** PLOT ROUTINE
   30 CONTINUE
C
C-------- SET UP FOR PLOT
C
      IDIF=II2-II1+1
      JDIF=JJ2-JJ1+1
      NM=1
      DO 100 N=1,IDIF
      DO 100 M=1,JDIF
      I1=II1+N-1
      J1=JJ1+M-1
      P(1,NM)=R(J1)
      P(2,NM)=Z(I1)
      P(3,NM)=V(I1,J1)
      ICON(NM)=10
      IF(M.NE.1) ICON(NM)=0
      NM=NM+1
  100 CONTINUE
      DO 200 M=1,JDIF
      DO 200 N=1,IDIF
      J1=JJ1+M-1
      I1=II1+N-1
      P(1,NM)=R(J1)
      P(2,NM)=Z(I1)
```

```
      P(3,NM)=V(I1,J1)
      ICON(NM)=10
      IF(N.NE.1) ICON(NM)=0
      NM=NM+1
200   CONTINUE
      NUMAX=3000
      NUM=NM-1
      CALL POLSUR(JDIF,IDIF)
      DO 150 MM=1,JDIF
      M=JJ1+MM-1
      NUM=NUM+1
      P(1,NUM)=R(M)
      P(2,NUM)=Z(II1)
      P(3,NUM)=0.0
      ICON(NUM)=10
      NUM=NUM+1
      P(1,NUM)=R(M)
      P(2,NUM)=Z(II1)
      P(3,NUM)=V(II1,M)
      ICON(NUM)=0
150   CONTINUE
      DO 160 MM=1,JDIF
      M=JJ1+MM-1
      NUM=NUM+1
      P(1,NUM)=R(M)
      P(2,NUM)=Z(II2)
      P(3,NUM)=0.0
      ICON(NUM)=10
      NUM=NUM+1
      P(1,NUM)=R(M)
      P(2,NUM)=Z(II2)
      P(3,NUM)=V(II2,M)
      ICON(NUM)=0
160   CONTINUE
      DO 170 NN=1,IDIF
      NUM=NUM+1
      N=NN+II1-1
      P(1,NUM)=R(JJ2)
      P(2,NUM)=Z(N)
      P(3,NUM)=0.0
      ICON(NUM)=10
      NUM=NUM+1
      P(1,NUM)=R(JJ2)
      P(2,NUM)=Z(N)
      P(3,NUM)=V(N,JJ2)
      ICON(NUM)=0
170   CONTINUE
      NUM=NUM+1
      P(1,NUM)=R(JJ2)
      P(2,NUM)=Z(II3)
      P(3,NUM)=V(II3,JJ2)
      P(4,NUM)=11.
      ICON(NUM)=31
      NUM=NUM+1
      P(1,NUM)=R(JJ1)
      P(2,NUM)=Z(II1)-RGZ*0.25
      P(3,NUM)=RGV*0.5
```

```
      P(4,NUM)=LA(3)
      ICON(NUM)=32
      NUM=NUM+1
      P(1,NUM)=R(JJ2)+RGR*0.1
      P(2,NUM)=Z(II1)+RGZ*0.5
      P(3,NUM)=0.0
      P(4,NUM)=LA(1)
      ICON(NUM)=32
      NUM=NUM+1
      P(1,NUM)=R(JJ1)+RGR*0.5
      P(2,NUM)=Z(II1)-RGZ*0.1
      P(3,NUM)=0.0
      P(4,NUM)=LA(2)
      ICON(NUM)=32
      CALL SYMCON(.07,4,-1.1,-1.2)
      NUM=NUM+1
      P(1,NUM)=R(JJ2)
      P(2,NUM)=Z(II1)
      P(3,NUM)=0.0
      P(4,NUM)=R(JJ2)
      ICON(NUM)=33
C--------  X-AXIS AT Y=Z(II1)
      RP(1)=JDIF
      DO 300 KK=1,JDIF
      IJ=KK*2
      JK=JJ1+KK-1
      RP(IJ)=R(JK)
      IJ=IJ+1
      RP(IJ)=-1
  300 CONTINUE
      PRINT 398
  398 FORMAT(10X,'   R-AXIS')
      PRINT 399,(RP(LL),LL=1,IJ)
  399 FORMAT(5X,10F10.4)
      PT(1)=R(JJ1)
      PT(2)=Z(II1)
      PT(3)=0
      LAB=1
      CALL AXES(RP,PT,LAB,2,1)
C--------  X-AXIS AT Y=Z(II2)
      PT(1)=R(JJ1)
      PT(2)=Z(II2)
      PT(3)=0
      LAB=1
      CALL AXES(RP,PT,LAB,2,2)
      CALL SYMCON(0.07,4,1.1,-1.2)
C--------  YAXIS AT X=R(JJ1)
      RP(1)=IDIF
      DO 301 KK=1,IDIF
      IJ=KK*2
      JK=II1+KK-1
      RP(IJ)=Z(JK)
      IJ=IJ+1
      RP(IJ)=-1
  301 CONTINUE
      PRINT 397
  397 FORMAT(10X,'   Z-AXIS')
```

```
      PRINT 399,(RP(LL),LL=1,IJ)
      PT(1)=R(JJ1)
      PT(2)=Z(II1)
      PT(3)=0
      LAB=2
      CALL AXES(RP,PT,LAB,2,2)
C---------  Y-AXIS AT TOP OF V
      PT(1)=R(JJ1)
      PT(2)=Z(II1)
      PT(3)=RGV
      LAB=2
      CALL AXES(RP,PT,LAB,2,1)
C--------  Y-AXIS AT X=R(JJ2)
      PT(1)=R(JJ2)
      PT(2)=Z(II1)
      PT(3)=0
      RP(3)=1
      RP(IJ)=1
      LAB=2
      CALL AXES(RP,PT,LAB,2,1)
C--------  Z-AXIS AT X=R(JJ1), Y=Z(II1)
      CALL SYMCON(0.07,1,-1.1,1.2)
      RP(1)=RGV+1.
      RR(1)=RGV+1
      II=RR(1)+1
       DO 302 KK=1,II
      IJ=KK*2
      RP(IJ)=KK-1
      RR(IJ)=KK-1
      IJ=IJ+1
      RR(IJ)=(-1)**(KK+1)
      RP(IJ)=-1.
  302 CONTINUE
      PRINT 396
  396 FORMAT(10X,'    V-AXIS')
      PRINT 399,(RR(LL),LL=1,IJ)
      PT(1)=R(JJ1)
      PT(2)=Z(II1)
      PT(3)=0
      LAB=3
      CALL AXES(RR,PT,LAB,2,1)
      PT(1)=R(JJ1)
      PT(2)=Z(II2)
      PT(3)=0.
      LAB=3
      CALL AXES(RP,PT,LAB,2,1)
      PRINT 400
  400 FORMAT(1H0,3X,37HTHREE DIMENSIONAL POINTS IN PLOT FILE/1H0,6X,
     15HPOINT,23X,1HR,14X,1HZ,14X,1HV)
      DO 299 LL=1,NUM
      PRINT 199,LL,ICON(LL),P(1,LL),P(2,LL),P(3,LL)
  199 FORMAT(5X,I5,5X,I5,5X,3F15.4)
  299 CONTINUE
C
C--------  END OF PLOT SETUP
C
      WRITE(6,34)
```

```
34 FORMAT(1H0,3X,43HSUMMARY OF ADDITIONAL SCIENTIFIC INPUT DATA)
   WRITE(6,35)WAVEL,NPULSE
35 FORMAT(1H0,11HWAVELENGTH=,E9.4,2HNM,8X,17HNUMBER OF PULSES=,I5)
   WRITE(6,36)DPULSE,RIM
36 FORMAT(1H0,12HPULSE WIDTH=,E9.4,3HSEC,10X,13HIMAGE RADIUS=,E9.4,
  12HCM)
   WRITE(6,37)REPET
37 FORMAT(1H0,16HREPETITION RATE=,E9.4,10HPULSES/SEC)
C
   WRITE(6,40)
40 FORMAT(1H0,17HAXIAL DISTANCE,CM)
   WRITE(6,41)
41 FORMAT(1H0,18HRADIAL DISTANCE,CM)
   WRITE(6,42)
42 FORMAT(1H0,25HTEMPERATURE RISE,DEGREE C)
   WRITE(6,43) TIMEX,NRUN
43 FORMAT(1H0,27HTEMPERATURE RISE PROFILE AT,E9.4,9HSEC (RUN=,I4,1H))
   WRITE(6,44)
44 FORMAT(1H0,3X,17HPLOT COMMAND LIST/)
   CALL PLOT(12.,-11.,-3)
   CALL PLOT(0.,.5,-3)
   HT=.14
   A=TIMEX
   B=NRUN
   CALL SYMBOL(0.,1.,HT,29H TEMPERATURE RISE PROFILE AT ,0.,29)
   XX=29*HT
   CALL FNUM(XX,1.,A,12)
   XX=XX+16*HT
   CALL SYMBOL(XX,1.,HT,13HSEC -- RUN = ,0.0,13)
   XX=XX+13*HT
   CALL NUMBER(XX,1.,HT,B,0.0,0)
   IF(SFAC.EQ.0.)GO TO 45
   FPN=SFAC
   IF(SFLAG.EQ.0.)FPN=1./SFAC
   CALL SYMBOL(0.,.75,.1,29H ORIGINAL T,C = PLOTTED T,C* ,0.,29)
   CALL NUMBER(2.9,.75,.1,FPN,0.,5)
45 CALL READIN(IRR)
   IF(IRR.EQ.1)GO TO 50
   GO TO 5
50 CALL PLOT(12.,0.,999)
   STOP
   END
   SUBROUTINE POLSUR(M,N)
   COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX
   DIMENSION W(3,500)
   NCT=0
   DO 10 I=1,N
   DO 10 J=1,M
   NCT=NCT+1
   DO 10 L=1,3
   W(L,NCT)=P(L,NCT)
10 CONTINUE
   NUM=0
   DO 20 N1=1,N
   NLO=N1
   MM=M-1
   DO 20 M1=1,MM
```

```
      MLO=M1
      NUM=NUM+1
      NA=M1+N1*M-M
      CALL EQUIV(P(1,NUM),W(1,NA))
      ICON(NUM)=0
      IF(M1.EQ.1) ICON(NUM)=10
      NUM=NUM+1
      NA=M1+1+(N1-1)*M
      CALL EQUIV(P(1,NUM),W(1,NA))
      ICON(NUM)=0
      NUM=NUM+1
      ISIGN=1
      IF(N1.NE.1) ISIGN=-1
      NA=MLO+(NLO-1)*M+1
      NB=NA-1
      NC=NA+ISIGN*M
      ISIGN=-ISIGN
      CALL PCROSS(W(1,NA),W(1,NB),W(1,NC),P(1,NUM),ISIGN)
      ICON(NUM)=50
   20 CONTINUE
      DO 30 M1=1,M
      MLO=M1
      NN=N-1
      DO 30 N1=1,NN
      NLO=N1
      NUM=NUM+1
      NA=M1+(N1-1)*M
      CALL EQUIV(P(1,NUM),W(1,NA))
      ICON(NUM)=0
      IF(N1.EQ.1) ICON(NUM)=10
      NUM=NUM+1
      NA=M1+(N1-1)*M+M
      CALL EQUIV(P(1,NUM),W(1,NA))
      ICON(NUM)=0
      NUM=NUM+1
      ISIGN=1
      IF(M1.EQ.M) ISIGN=-1
      NA=MLO+(NLO-1)*M+M
      NB=NA+ISIGN
      NC=NA-M
      ISIGN=-ISIGN
      CALL PCROSS(W(1,NA),W(1,NB),W(1,NC),P(1,NUM),ISIGN)
      ICON(NUM)=50
   30 CONTINUE
      RETURN
      END
      SUBROUTINE PCROSS(PA,PB,PC,V,IS)
      DIMENSION PA(3),PB(3),PC(3),V(3)
      DIMENSION VX(3),VY(3)
      DO 10 I=1,3
      VX(I)=PB(I)-PA(I)
      VY(I)=PC(I)-PA(I)
   10 CONTINUE
      V(1)=VX(2)*VY(3)-VX(3)*VY(2)
      V(2)=-(VX(1)*VY(3)-VX(3)*VY(1))
      V(3)=VX(1)*VY(2)-VX(2)*VY(1)
      SUM=0.0
```

```
      DO 20 I=1,3
   20 SUM=SUM+V(I)*V(I)
      SUM=SQRT(SUM)+1.0E-20
      DO 30 I=1,3
   30 V(I)=IS*V(I)/SUM
      RETURN
      END
      SUBROUTINE EQUIV(PA,PB)
      DIMENSION PA(3),PB(3)
      DO 10 I=1,3
   10 PA(I)=PB(I)
      RETURN
      END
      SUBROUTINE SYMCON(HH,NN,XX,YY)
      COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX
      NUM=NUM+1
      DO 10 I=1,3
   10 P(I,NUM)=0.0
      P(4,NUM)=HH
      ICON(NUM)=71
      NUM=NUM+1
      DO 20 I=1,3
   20 P(I,NUM)=0.0
      P(4,NUM)=NN
      ICON(NUM)=72
      NUM=NUM+1
      DO 30 I=1,3
   30 P(I,NUM)=0.0
      P(4,NUM)=XX
      ICON(NUM)=73
      NUM=NUM+1
      DO 40 I=1,3
   40 P(I,NUM)=0.0
      P(4,NUM)=YY
      ICON(NUM)=74
      RETURN
      END
      SUBROUTINE READIN(IRR)
      COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX
      COMMON/PLBAS2/ AP(16),AV(16),CP(16),DAT(8)
      COMMON/PLBAS3/ WINXL,WINYL,WINXW,WINYW,IWIN
      COMMON/PLBAS4/ SCRNXL,SCRNYL,SCRNXW,SCRNYW,ISCRN
      COMMON/PLBAS5/ SIGNOR,SNPLOT,IH
      DIMENSION NAM(21)
      DATA NAM/ 4HP   ,4HINIT,4HROLL,4HPITC,4HYAW ,
     X          4HSCAL,4HTRAN,4HDIST,4HREIN,4HHIDE,
     X          4HSIGN,4HWIND,4HSCRN,4HBOX ,4HFACT,
     X          4HPLOT,4HUSER,4HPRIN,4HEND ,4HDUM ,
     X          4HAXIS /
      DATA NONAM/21/
      EQUIVALENCE (DAT(1),RDAR(1))
      DIMENSION RDAR(8)
      DIMENSION R(4),RMN(3),RMX(3),PT(3)
      IPRIN=0
    1 READ(5,10,END=999) NAMM,(RDAR(L),L=2,8)
   10 FORMAT(A4,6X,7F10.4)
      IF(IPRIN.GT.0) GO TO 41
```

```
      WRITE(6,40) NAMM,(RDAR(L),L=2,8)
   40 FORMAT(1X,A4,6X,7F10.4)
   41 CONTINUE
C---  COMPARE TO PRESTORED NAMES IN ORDER TO DETERMINE THE ACTION CODE
      DO 20 I=1,NONAM
      IF(NAMM.EQ.NAM(I)) GO TO 30
   20 CONTINUE
C---  ERROR PATH -- INPUT WORD WAS NOT VALID
      IRR=1
      WRITE(6,50) NAMM,NAM
   50 FORMAT(/,' ERROR -- THE CODE NAME ',A5,1X,' WAS NOT VALID,  VALID
     X NAMES ARE AS FOLLOWS',/,20(1X,A4))
      GO TO 999
   30 CONTINUE
      IF(I.EQ.1) GO TO 100
      IF(I.GT.1.AND.I.LT.17) GO TO 120
      IK=I-16
      GO TO (170,180,190,200,210),IK
  100 CONTINUE
      IF(RDAR(2).LT.-0.1.OR.RDAR(2).GT.99.) GO TO 110
      NUM=NUM+1
      DO 111 L=1,4
  111 P(L,NUM)=RDAR(L+2)
      ICON(NUM)=RDAR(2)
      GO TO 1
  110 CONTINUE
      NUM=RDAR(3)
      GO TO 1
  120 CONTINUE
      RDAR(1)=I-1
      CALL SSPLOT
      GO TO 1
  170 CONTINUE
      RDAR(1)=17
      CALL USER
      GO TO 1
  180 CONTINUE
      IPRIN=RDAR(2)
      GO TO 1
  190 CONTINUE
      GO TO 999
  200 CONTINUE
      WRITE(6,201) NUM,NUMAX
  201 FORMAT(5X,'CURRENT NUMBER OF POINTS= ',I6,' AND MAXIMUM ALLOWED= '
     X,I6)
      NUM1=MINO(NUMAX,NUM)
      IF(NUM1.LE.0) GO TO 1
      WRITE(6,205)
  205 FORMAT( 1X,10HCOORDINATE,10H  LOW VAL  ,10H  HI VAL    ,
     X 10H  MEAN VAL   ,10H  WIDTH      )
      DO 202 J=1,3
      RMIN=1.0E+20
      RMAX=-1.0E+20
      DO 203 L=1,NUM
      IF(ICON(L).GE.49) GO TO 203
      RMIN=AMIN1(RMIN,P(J,L))
      RMAX=AMAX1(RMAX,P(J,L))
```

```
 203 CONTINUE
     RMEAN=(RMAX+RMIN)/2.0
     DIF=RMAX-RMIN
     WRITE(6,204) J,RMIN,RMAX,RMEAN,DIF
 204 FORMAT(1X,'COORD ',I2,1X,4F10.3)
 202 CONTINUE
     GO TO 1
 210 CONTINUE
     IF(NUM.LE.0) GO TO 999
     DO 211 J=1,3
     RMIN=1.0E+20
     RMAX=-RMIN
     DO 212 I=1,NUM
     IF(ICON(I).GT.49) GO TO 212
     RMIN=AMIN1(RMIN,P(J,I))
     RMAX=AMAX1(RMAX,P(J,I))
 212 CONTINUE
     RMX(J)=RMAX
     RMN(J)=RMIN
     PT(J)=(RMIN+RMAX)/2.0
     IF(RDAR(2).GT.0.1) PT(J)=RMAX
     IF(RDAR(2).LT.-0.1) PT(J)=RMIN
 211 CONTINUE
     DO 213 J=1,3
     IF(RMX(J)-RMN(J).LT.0.0001) GO TO 213
     IF(RMX(J)-RMN(J).GT.1.0E+20) GO TO 213
     R(1)=RMN(J)
     R(2)=(RMX(J)-RMN(J))/5.0
     R(3)=6.0
     R(4)=5.0
     LAB=J
     CALL AXES(R,PT,LAB,1)
 213 CONTINUE
     GO TO 1
 999 CONTINUE
     RETURN
     END
     SUBROUTINE AXES(R,PT,LAB,MODE,NCON)
     COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX
     DIMENSION R(1),T(102)
     DATA NT/100/
     DIMENSION PT(3)
     DATA BIG/1.0E+20/
C---  OBJECTIVE OF ROUTINE IS TO GENERATE AXIS DATA IN THE THREE
C---  DIMENSIONAL POINT DATA BASE
C---  INPUT IS THRU CALLING ARGUMENTS AS FOLLOWS
C--- LAB SHOULD BE 1 2 OR 3 DENOTING X, Y OR Z AXIS INFORMATION
C--- IF MODE IS 1 THEN R(1,2,3 AND 4) DENOTE RESPECTIVELY THE START,
C---       INCREMENT,NUMBER OF INCREMENTS AND INCREMENT FOR NUMBERING
C--- MODE=2 MEANS THAT THE TICK DATA IS STORED IN THE ARRAY R SO THAT
C---  R(1) IS THE NUMBER OF POINTS, R(2) IS THE VALUE FOR THE FIRST  ,
C---  MARK, R(3) IS POSITIVE IF A NUMBER SHOULD BE PLOTTED, AND NEGATIVE
C---  OTHERWISE  AND SO ON
C---  IN THE CASE OF EACH MODE, TICK DATA IS BUILT INTO THE LOCAL ARRAY
C---  T AS A BUFFER, AND THEN TRANSFERRED TO THE POINT ARRAY
     GO TO (10,20),MODE
  10 CONTINUE
```

```
      START=R(1)
      AINC=R(2)
      NO=R(3)
      IVINC=R(4)
      IRR=1
      IF(NO.LE.0) GO TO 998
      IRR=2
      IF(NO.GT.NT/2) GO TO 998
      T(1)=NO
      SMIN=BIG
      SMAX=-BIG
      DO 11 I=1,NO
      T(2*I)=START+(I-1)*AINC
      T(2*I+1)=-1
      SMIN=AMIN1(T(2*I),SMIN)
      SMAX=AMAX1(T(2*I),SMAX)
      IF(IVINC.LE.0) GO TO 10
      IF(MOD(I,IVINC).EQ.1) T(2*I+1)=1.0
   11 CONTINUE
      GO TO 100
   20 CONTINUE
      NO=R(1)
      IRR=3
      IF(NO.LE.0) GO TO 998
      IRR=4
      IF(NO.GT.NT/2) GO TO 998
      SMIN=BIG
      SMAX=-BIG
      DO 21 I=1,NO
      T(2*I)=R(2*I)
      T(2*I+1)=R(2*I+1)
   21 CONTINUE
  100 CONTINUE
      JTEM=NUM
      DO 110 I=1,NO
      JTEM=JTEM+1
      DO 120 J=1,3
  120 P(J,JTEM)=PT(J)
      P(4,JTEM)=LAB
      P(LAB,JTEM)=T(2*I)
      IF(I.EQ.1) ICON(JTEM)=NCON*10+1
      IF(I.NE.1) ICON(JTEM)=1
  110 CONTINUE
      NUM=NUM+NO
      JTEM=NUM
      DO 130 I=1,NO
      IF(T(2*I+1).LT.0.0) GO TO 130
      NUM=NUM+1
      JTEM=JTEM+1
      DO 140 J=1,3
  140 P(J,JTEM)=PT(J)
      P(LAB,JTEM)=T(2*I)
      ICON(JTEM)=33.
      P(4,JTEM)=T(2*I)
  130 CONTINUE
  999 IRR=0
      RETURN
```

```
  998 WRITE(6,997) IRR
  997 FORMAT(/,'    ERROR IN AXES ROUTINE, IRR= ',I6,/)
      RETURN
      END
      SUBROUTINE SSPLOT
      COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX
      COMMON/PLBAS2/ AP(16),AV(16),CP(16),DAT(8)
      COMMON/PLBAS3/ WINXL,WINYL,WINXW,WINYW,IWIN
      COMMON/PLBAS4/ SCRNXL,SCRNYL,SCRNXW,SCRNYW,SCRNZW,ISCRN
      COMMON/PLBAS5/ SIGNOR,SNPLOT,IH
      COMMON/PLBAS6/DIMAGE,DORIG,DOBX,DOBY
      COMMON/PLBAS7/HT,NDECFX,XLATE,YLATE
C---    AP,AV ARE PROJECTIVE NON SINGULAR MATRICES WHICH RECORD THE
C---    CURRENT POSITION OF THE POINT SET
C---   IH  THE HIDDEN LINE FLAG
C---   ZVIEW IS DISTANCE OF VIEWERS EYE FROM PROJECTION(XY) PLANE
C---   DAT CONTAINS THE COMMAND DATA FOR EXECUTING PIECES OF THIS ROUTINE
C---   SIGNOR  THE SIGN APPLIED TO THE SURFACE NORMALS
C---   P CONTAINS XYZ DATA OF POINTS,VECTORS AND SYMBOL DATA IN 4TH PLC
C---   ICON CONTAINS TWO PACKED DIGITS  AB WITH THE FOLLOWING MEANING
C--- A=0, CONTINUE PRESENT MODE OF PLOTTING, A=1 START CONNECTING POINTS
C---   BY STRAIGHT LINES, A=2 CONNECT PTS BY DASHED LINES, A=4 PLOT POINT
C---   S ONLY, A=4 PLOT DASHED POINTS
C--- B=0 PLOT NO SYMBOL, B=1 PLOT CENTERED SYMBOL WHOSE VALUE IS P(4,)
C---   PLOT LITERAL STRING IN FIELD P(4,) B=3 PLOT NUMBER IN FIELD P(4,)
C--- SET UP WINDOW PARAMETERS
      DATA SMALL/1.0E-10/,SMAL/1.0E-8/
      DIMENSION AID(16),TP(16),BP(16)
      DIMENSION RWID(3),RCEN(3),RMIN(3),RMAX(3)
      DIMENSION PP(3),VV(3)
      DATA AID/1.0,4*0.0,1.0,4*0.0,1.0,4*0.0,1.0/
      IT=DAT(1)
      GO TO (10,20,30,40,50,60,70,80,90,100,110,120,130,140,150),IT
C---   IT=1  INITIALIZE KEY VARIABLES WITH DEFAULT VALUES
   10 SIGNOR=1.0
      NUM=0
      IPRIN=0
      HT=0.07
      SWIDTH=8.25
      SHEIGT=6.5
      ISCRN=-1
      IWIN=-1
      SCRNXL=0.0
      SCRNYL=0.0
      SCRNXW=8.5
      SCRNYW=6.25
      SCRNZW=SCRNXW
      SXUNIT=1024.
      SYUNIT=760.0
      IH=0
      ZVIEW=0.0
      NERASE=0
      NDECFX=-1
      XLATE=-1.1
      YLATE=-1.1
      IF(IPLTX.GT.0)GO TO 12
      CALL PLOTS(0,0,8)
```

```
      IPLTX=1
   12 DO 11 I=1,16
      BP(I)=AID(I)
      AP(I)=AID(I)
   11 AV(I)=AID(I)
      BP(11)=0.0
C--- REPLACE INCREMENTAL VALUES WITH ABSOLUTE VALUES
      NUMAX=3000
      DO 13 L=1,NUMAX
      DO 14 K=1,4
   14 P(K,L)=0.0
      ICON(L)=0
   13 CONTINUE
      DOBX=0.0
      DOBY=0.0
      GO TO 999
C--- 20,30 AND 40 ARE ROTATION COMMANDS
C--- IT=2  XYROT OR ROLL
   20 DAT(1)=1.0
      CALL PERSPT(DAT,TP)
      CALL MMULT(AP,TP,CP,1)
      CALL MMULT(AV,TP,CP,1)
      GO TO 999
C--- IT=3 YZROT OR PITCH
   30 DAT(1)=2.0
      CALL PERSPT(DAT,TP)
      CALL MMULT(AP,TP,CP,1)
      CALL MMULT(AV,TP,CP,1)
      GO TO 999
C--- IT=4 ZXROT OR YAW
   40 DAT(1)=3
      CALL PERSPT(DAT,TP)
      CALL MMULT(AP,TP,CP,1)
      CALL MMULT(AV,TP,CP,1)
      GO TO 999
C--- IT=5 SCALE
   50 DAT(1)=4
      CALL PERSPT(DAT,TP)
      CALL MMULT(AP,TP,CP,1)
      GO TO 999
C--- IT=6 TRANSLATION
   60 DAT(1)=5
      CALL PERSPT(DAT,TP)
      CALL MMULT(AP,TP,CP,1)
      GO TO 999
C--- IT=7 SETUP PROJECTION ONTO XYPLAN FROM VIEWERS POSITION
   70 DAT(1)=6
      ZVIEW=DAT(2)
      DIMAGE=DAT(2)
      DORIG=DAT(3)
      DOBX=DAT(4)
      DOBY=DAT(5)
      CALL PERSPT(DAT,BP)
      GO TO 999
C---  REIDENTIFY THE TRANSFORMATION MATRICES
   80 DO 81 I=1,16
      AP(I)=AID(I)
```

```
   81 AV(I)=AID(I)                                                          PLT00742
      GO TO 999                                                             PLT00743
C---  SETUP THE HIDDEN LINE FLAG                                            PLT00744
   90 IH=DAT(2)                                                             PLT00745
      GO TO 999                                                             PLT00746
  100 SIGNOR=DAT(2)                                                         PLT00747
      GO TO 999                                                             PLT00748
  110 CONTINUE                                                              PLT00749
      IWIN=-1                                                               PLT00750
      IF(DAT(2)**2+DAT(3)**2+DAT(4)**2+DAT(5)**2.LT.SMAL) GO TO 999         PLT00751
      IWIN=1                                                                PLT00752
      WINXL=DAT(2)                                                          PLT00753
      WINYL=DAT(3)                                                          PLT00754
      WINXW=DAT(4)                                                          PLT00755
      WINYW=DAT(5)                                                          PLT00756
      GO TO 999                                                             PLT00757
C---  SCREEN PARAMETERS INTRODUCED                                          PLT00758
  120 CONTINUE                                                              PLT00759
      ISCRN=-ISCRN                                                          PLT00760
      IF(DAT(2)**2+DAT(3)**2+DAT(4)**2+DAT(5)**2.LT.SMAL) GO TO 999         PLT00761
      SCRNXL=DAT(2)                                                         PLT00762
      SCRNYL=DAT(3)                                                         PLT00763
      SCRNXW=DAT(4)                                                         PLT00764
      SCRNYW=DAT(5)                                                         PLT00765
      SCRNZW=DAT(6)                                                         PLT00766
      ISCRN=1                                                               PLT00767
      GO TO 999                                                             PLT00768
C---  BOX COMMAND, SCALE THE OBJECT TO FILL THE SCREEN                      PLT00769
  130 CONTINUE                                                              PLT00770
      IF(ISCRN.LT.0) GO TO 999                                              PLT00771
      PROA=DAT(2)                                                           PLT00772
      PROB=DAT(3)                                                           PLT00773
      PROC=DAT(4)                                                           PLT00774
C---  DETERMINE THE XYZ EXTENT OF THE TRANSFORMED OBJECT                    PLT00775
      DO 131 L=1,3                                                          PLT00776
      RMIN(L)=1.0E+20                                                       PLT00777
  131 RMAX(L)=-1.0E+20                                                      PLT00778
      I=0                                                                   PLT00779
  137 I=I+1                                                                 PLT00780
      IF(I.GT.NUM) GO TO 138                                                PLT00781
      CALL DECOD(PP,VV,AA,JCON,ISYM,IVEC,I)                                 PLT00782
      IF(IVEC.EQ.999) GO TO 137                                             PLT00783
      IF(I.LT.0) GO TO 999                                                  PLT00784
      WW=PP(1)*AP(13)+PP(2)*AP(14)+PP(3)*AP(15)+AP(16)+SMALL                PLT00785
      DO 132 L=1,3                                                          PLT00786
      L4=L*4                                                                PLT00787
      PPP=(PP(1)*AP(L4-3)+PP(2)*AP(L4-2)+PP(3)*AP(L4-1)+AP(L4))/WW          PLT00788
      RMIN(L)=AMIN1(PPP,RMIN(L))                                            PLT00789
      RMAX(L)=AMAX1(PPP,RMAX(L))                                            PLT00790
  132 CONTINUE                                                              PLT00791
      GO TO 137                                                             PLT00792
  138 CONTINUE                                                              PLT00793
      DO 133 L=1,3                                                          PLT00794
      RCEN(L)=(RMIN(L)+RMAX(L))/2.0                                         PLT00795
      RWID(L)=RMAX(L)-RMIN(L)+SMALL                                         PLT00796
      DAT(L+1)=-RCEN(L)                                                     PLT00797
  133 CONTINUE                                                              PLT00798
```

```
C---   CENTERISE THE OBJECT AROUND THE ORIGIN
       DAT(1)=5
       CALL PERSPT(DAT,TP)
       CALL MMULT(AP,TP,CP,1)
C---   SCALE THE OBJECT INTO THE SCREEN AREA OR WINDOW AREA IF REQUESTED
       IF(ISCRN.LE.0) GO TO 999
       A=1.0E+20
       SX=SCRNXW/RWID(1)*PROA
       IF(PROB.GT.0.0) GO TO 135
       A=SCRNYW/RWID(2)*PROA
       SX=AMIN1(SX,A)
       DAT(2)=SX
       DAT(3)=SX
       DAT(4)=SX
       GO TO 136
   135 CONTINUE
       SY=SCRNYW/RWID(2)*PROB
       DAT(2)=SX
       DAT(3)=SY
       DAT(4)=1.0
       IF(PROC.GT.0.0) DAT(4)=SCRNZW/RWID(3)*PROC
   136 CONTINUE
       DAT(1)=4
       CALL PERSPT(DAT,TP)
       CALL MMULT(AP,TP,CP,1)
       CALL MMULT(AV,TP,CP,1)
       IF(IWIN.LE.0) GO TO 999
C---   APPLY A FURTHER TRANSLATION AND SCALE IF WINDOW IS IN EFFECT
       DAT(1)=5.
       DAT(2)=-(WINXL+WINXW/2.0)
       DAT(3)=-(WINYL+WINYW/2.0)
       DAT(4)=0.0
       CALL PERSPT(DAT,TP)
       CALL MMULT(AP,TP,CP,1)
       DAT(2)=SCRNXW/WINXW
       DAT(3)=SCRNYW/WINYW
       DAT(2)=AMIN1(DAT(2),DAT(3))
       DAT(3)=DAT(2)
       DAT(4)=DAT(2)
       DAT(4)=1.0
       DAT(1)=4.0
       CALL PERSPT(DAT,TP)
       CALL MMULT(AP,TP,CP,1)
       CALL MMULT(AV,TP,CP,1)
       WINXW=SCRNXW
       WINYW=SCRNYW
       WINXL=SCRNXL
       WINYL=SCRNYL
       GO TO 999
C---   APPLY A STRAIGHT FACTOR TO ALL SUBSEQUENT PLTS
   140 CONTINUE
       IF(DAT(2).LE.SMAL) GO TO 999
       CALL FACTOR(DAT(2))
       GO TO 999
C--- MAIN PLOT PROCESSING IS HERE
   150 CONTINUE
       IF(DAT(4).LT.0.0) CALL PLOT(0.,0.,-3)
```

```
      IF(DAT(4).LT.0.0) CALL PLOT(DAT(2),DAT(3),999)
      IF(DAT(4).LT.0.0) GO TO 999
      CALL PLOT(DAT(2),DAT(3),-3)
  151 CONTINUE
      CALL MMULT(AP,BP,CP,3)
C---  SETUP THE WINDOW,SCREEN AND PLOT BOUNDARIES
      IF(IWIN.LE.0.AND.ISCRN.LE.0) GO TO 154
      IF(ISCRN.GT.0) GO TO 153
      IF(IWIN.LE.0) GO TO 154
      XL=WINXL
      YL=WINYL
      XW=WINXW
      YW=WINYW
      GO TO 152
  153 XL=SCRNXL
      YL=SCRNYL
      XW=SCRNXW
      YW=SCRNYW
  152 CONTINUE
      IF(DAT(2)**2+DAT(3)**2.GT.SMAL) CALL PLOT(XL+XW/2.0,YL+YW/2.0,3)
      CALL PLOT(XL,YL,3)
      CALL PLOT(XL+XW,YL,2)
      CALL PLOT(XL+XW,YL+YW,2)
      CALL PLOT(XL,YL+YW,2)
      CALL PLOT(XL,YL,2)
  154 CONTINUE
      MOVNOW=0
      IF(ISCRN.GT.0.OR.IWIN.GT.0) CALL WINDOW(XL,YL,XW,YW,MOVNOW)
      XLAS=0.0
      YLAS=0.0
      IPERM=0
      NPLT=0
      I=0
  301 I=I+1
      IF(I.GT.NUM) GO TO 302
C---  MAIN PLOTTING LOOP
      X1=XLAS
      Y1=YLAS
C---  DECODE THE NECESSARY POINT AND AUXILIARY DATA
      IA=I
      CALL DECOD(PP,VV,AA,JCON,ISYM,IVEC,IA)
      IF(IVEC.EQ.999) GO TO 301
      IF(IA.LE.0) GO TO 300
      I=IA
      IF(JCON*(5-JCON).NE.0) IPERM=JCON
      IF(IPERM.EQ.0) GO TO 300
      WNOW=PP(1)*CP(13)+PP(2)*CP(14)+PP(3)*CP(15)+CP(16)+SMALL
      XNOW=(PP(1)*CP(1)+PP(2)*CP(2)+PP(3)*CP(3)+CP(4))/WNOW
      YNOW=(PP(1)*CP(5)+PP(2)*CP(6)+PP(3)*CP(7)+CP(8))/WNOW
      X2=XNOW
      Y2=YNOW
      MOVNOW=2
      IF(IWIN.LT.0) GO TO 310
C---  MAKE THE REQUIRED WINDOW CHECK
      MOVNOW=1
      CALL WINDOW(X1,Y1,X2,Y2,MOVNOW)
  310 CONTINUE
```

```
      IF(MOVNOW.LT.0) GO TO 600
      IF(IH.EQ.0.OR.IVEC.LT.1) GO TO 320
C---  MAKE THE HIDDEN LINE/SURFACE NORMAL CHECK
      VXNOW=VV(1)*AV(1)+VV(2)*AV(2)+VV(3)*AV(3)
      VYNOW=VV(1)*AV(5)+VV(2)*AV(6)+VV(3)*AV(7)
      VZNOW=VV(1)*AV(9)+VV(2)*AV(10)+VV(3)*AV(11)
      PXNOW=PP(1)*AP(1)+PP(2)*AP(2)+PP(3)*AP(3)+AP(4)
      PYNOW=PP(1)*AP(5)+PP(2)*AP(6)+PP(3)*AP(7)+AP(8)
      PZNOW=PP(1)*AP(9)+PP(2)*AP(10)+PP(3)*AP(11)+AP(12)
      IF(ABS(BP(15)).LT.0.0001) GO TO 330
      ZVIEW=-BP(16)/BP(15)
      D=(PXNOW-DOBX)*VXNOW+(PYNOW-DOBY)*VYNOW+(PZNOW-ZVIEW)*VZNOW
      D=D*SIGNOR
      IHCUR=0
      PRINT 311
  311 FORMAT(' PXNOW,PYNOW,PZNOW,VXNOW,VYNOW,VZNOW,DOBX,DOBY,ZVIEW,D')
      WRITE(6,312) I,
     X          PXNOW,PYNOW,PZNOW,VXNOW,VYNOW,VZNOW,DOBX,DOBY,ZVIEW,D
  312 FORMAT(1X,I4,3(3(1X,F9.3)),2X,F9.3)
      IF(D.GT.0.0) IHCUR=1
      GO TO 340
  330 IHCUR=0
      D=VZNOW*SIGNOR
      IF(D.LT.0.0) IHCUR=1
  340 CONTINUE
  320 CONTINUE
      IPERMN =IPERM
      IF(IH.EQ.0.OR.IVEC.LT.1) GO TO 350
      IF(IHCUR.EQ.0) GO TO 350
      IF(IH.EQ.2) GO TO 360
C---  TOTALLY HIDDEN LINE
      IPERMN=0
      GO TO 350
  360 CONTINUE
      IF(IPERM.EQ.1) IPERMN =2
      IF(IPERM.EQ.2) IPERMN =4
  350 CONTINUE
      IF(IPERMN .EQ.0) GO TO 600
      IF((IPERMN-2)*(IPERMN-4).EQ.0.AND.JCON.EQ.0) GO TO 370
      NDASH=1
      UX=X2-X1
      UY=Y2-Y1
      GO TO 380
  370 CONTINUE
      D=SQRT((X2-X1)**2+(Y2-Y1)**2)
      NDASH=D/0.25
      NDASH=MAX0(3,NDASH)
      D1=D/NDASH
      UX=(X2-X1)/(D+SMALL)*D1
      UY=(Y2-Y1)/(D+SMALL)*D1
C---  POSITION POINT AT START OF SEGMENT
      IF(MOVNOW.EQ.3.OR.MOVNOW.EQ.5) CALL PLOT(X1,Y1,3)
      IF(MOVNOW.EQ.3.OR.MOVNOW.EQ.5) NPLT=NPLT+1
  380 CONTINUE
      IF(IPERMN .GT.2) GO TO 420
      MODO=-1
      DO 410 J=1,NDASH
```

```
      XX=X1+UX*J
      YY=Y1+UY*J
      MODO=-MODO
      IPLT=2
      IF(MODO.LT.0) IPLT=3
      IF(JCON.NE.0) IPLT=3
      NPLT=NPLT+1
      CALL PLOT(XX,YY,IPLT)
  410 CONTINUE
      GO TO 500
  420 DO 430 J=1,NDASH
      XX=X1+UX*J
      YY=Y1+UY*J
      CALL PLOT(XX,YY,3)
      CALL PLOT(XX,YY,2)
      NPLT=NPLT+1
  430 CONTINUE
      GO TO 500
  500 CONTINUE
      IF(MOVNOW.EQ.4.OR.MOVNOW.EQ.5) GO TO 590
      IF(ISYM.EQ.0) GO TO 590
      GO TO (510,520,530),ISYM
  510 CONTINUE
      INT=AA
      CALL SYMBOL(X2,Y2,HT,INT,0.0,-2)
      GO TO 590
  520 CONTINUE
      NCHAR=4.0
      XLEFT=(XLATE-1.0)*0.5*NCHAR*HT
      YLEFT=(YLATE-1.0)*0.5*NCHAR*HT
      CALL SYMBOL( X2+XLEFT,Y2+YLEFT,HT,AA,0.0,4)
      GO TO 590
  530 CONTINUE
      SZ=2
      S1=ABS(AA)
      IF(S1.GT.SMAL) SZ=ALOG10(S1)
      IF(S1.LT.0.0001) GO TO 591
      SZ=ALOG10(S1)
      IF(SZ.GE.0.0) NDEC=1
      IF(SZ.LT.0.0) NN=SZ
      IF(SZ.LT.0.0) NDEC=NN+2
      IF(NDECFX.GE.0) NDEC=NDECFX
      IF(SZ.GE.0.0) NSIG=SZ+1.0+2.0
      IF(SZ.LT.0.0) NSIG=NDEC+2.0
      GO TO 592
  591 CONTINUE
      NSIG=3
      NDEC=1
  592 CONTINUE
      IF(AA.LT.0.0) NSIG=NSIG+1
      XLEFT=HT*NSIG*(XLATE-1.0)*0.5
      YLEFT=HT*NSIG*(YLATE-1.0)*0.5
      CALL NUMBER(X2+XLEFT,Y2+YLEFT,HT,AA,0.0,NDEC)
  590 CONTINUE
      XLAS=XNOW
      YLAS=YNOW
      GO TO 300
```

```
  600 CONTINUE
  300 CONTINUE
      GO TO 301
  302 CONTINUE
      WRITE(6,390) NPLT
  390 FORMAT(6X,'PLOT COMPLETED, TOTAL POINTS PLOTTED= ',I6)
      GO TO 999
  999 RETURN
      END
      SUBROUTINE WINDOW(XA,YA,XB,YB,MOD)
C---  ROUTINE TO EXAMINE THE CURRENT SEGMENT RELATIVE TO THE CURRENT
C---  WINDOW
C---  INPUT  IF MOD IS 0 THEN XA,YA ARE LOWER LEFT CORNER OF NEW WINDOW
C---   AND XB AND YB ARE THE WIDTH AND HEIGHT OF THE WINDOW
C---  OTHER PARAMETERS ARE ALSO INITIALIZED IN THIS CASE
C---  THE RETURN VALUE OF MOD IS -1
C---  IF MOD IS 1 THEN XA,YA AND XB,YB REPRESENT END POINTS OF A LINE
C---  SEGMENT WHICH SHOULD BE WINDOWED. IF MOD=-1 ON RETURN THE SEGMENT
C---  DOES NOT INTERSECT THE WINDOW, IF MOD=2 THE INTERSECTION OCCURS
C---  AND THE FIRST POINT DOES NOT CHANGE, WHILE IF MOD=3 THE FIRST P
C---  HAS CHANGED.  XA,YA,XB,YB MAY BE MODIFIED ON OUTPUT TO HOLD
C---  CHANGED VALUES OF THE END POINTS
C---  IF MOD IS LESS THAN -1, AN ERROR HAS OCCURRED
      DIMENSION PX(2),PY(2),PD(5),X(5),Y(5),IND(2,2)
      DATA IND/1,2,4,3/
      DATA SMAL/1.0E-20/
      LOGICAL AIN,BIN
      BET(A,B,C)=(B-A)*(C-B)
      IF(MOD)20,10,20
C---  INITIALIZATION OF WINDOW PARAMETERS
   10 CONTINUE
      XL=XA
      YL=YA
      XW=XB
      YW=YB
      XU=XL+XW
      YU=YL+YW
      X(1)=XL
      X(2)=XL+XW
      X(3)=X(2)
      X(4)=XL
      X(5)=XL
      Y(1)=YL
      Y(2)=YL
      Y(3)=YL+YW
      Y(4)=Y(3)
      Y(5)=YL
      HXW=XW/2.0
      HYW=YW/2.0
      XC=XL+HXW
      YC=YL+HYW
      DC=HXW*HXW+HYW*HYW
      MOD=-1
      GO TO 999
C---  BEGIN WINDOW CUTTING ACTION ON SEGMENT
   20 CONTINUE
      AX=BET(XL,XA,XU)
```

```
      AY=BET(YL,YA,YU)
      AIN=.TRUE.
      IF(AX.LT.0.0.OR.AY.LT.0.0) AIN=.FALSE.
      BX=BET(XL,XB,XU)
      BY=BET(YL,YB,YU)
      BIN=.TRUE.
      IF(BX.LT.0.0.OR.BY.LT.0.0) BIN=.FALSE.
      IF(AIN.AND.BIN) GO TO 100
      IF(AIN.OR.BIN) GO TO 200
      GO TO 300
C---  BOTH INSIDE
  100 CONTINUE
      MOD=2
      GO TO 999
C---  ONE INSIDE/ ONE OUTSIDE
  200 CONTINUE
      IF(AIN) GO TO 210
      XX=XA
      YY=YA
      GO TO 220
  210 XX=XB
      YY=YB
  220 CONTINUE
C---  CHOOSE THE MAIN CORNER REFERENCE POINT
      SX=XX-XC
      SY=YY-YC
      I=2
      J=2
      IF(SX.LT.0.0) I=1
      IF(SY.LT.0.0) J=1
      IS=IND(I,J)
C---  SET UP THE EQN OF THE LINE SEGMENT
      A=YB-YA
      B=XA-XB
      C=XB*YA-XA*YB
      ISA=IS-1
      IF(ISA.LT.1) ISA=4
      D1=A*X(IS)+B*Y(IS)+C
      D2=A*X(ISA)+B*Y(ISA)+C
      IF(D1*D2.GT.0.0) ISA=IS+1
      IF(ISA.GT.4) ISA=1
      ICUM=ISA+IS
      IF(ICUM.NE.5) GO TO 240
      XX=X(IS)
      YY=-(C+A*X(IS))/(B+SMAL)
      GO TO 250
  240 XX=-(C+B*Y(IS))/(A+SMAL)
      YY=Y(IS)
  250 CONTINUE
      IF(AIN) GO TO 260
      XA=XX
      YA=YY
      MOD=3
      GO TO 999
  260 CONTINUE
      XB=XX
      YB=YY
```

```
      MOD=4
      GO TO 999
C---  THE CASE OF TWO POINTS OUTSIDE THE WINDOW
  300 CONTINUE
      IF(XA-XL.LT.0.0.AND.XB-XL.LT.0.0) GO TO 390
      IF(XA-XU.GT.0.0.AND.XB-XU.GT.0.0) GO TO 390
      IF(YA-YL.LT.C.0.AND.YB-YL.LT.0.0) GO TO 390
      IF(YA-YU.GT.0.0.AND.YB-YU.GT.0.0) GO TO 390
      A=YB-YA
      B=XA-XB
      C=XB*YA-XA*YB
      ICUM=0
      PD(1)=A*X(1)+B*Y(1)+C
      DO 310 I=2,5
      PD(I)=A*X(I)+B*Y(I)+C
      IF(PD(I)*PD(I-1).LT.0.0) ICUM=ICUM+1
  310 CONTINUE
      IF(ICUM.EQ.0) GO TO 390
      NUM=0
      DO 340 I=1,4
      IF(PD(I)*PD(I+1).GT.0.0) GO TO 340
      NUM=NUM+1
      IF(NUM.GT.2) GO TO 340
      ICUM=I+I+1
      IF(ICUM.EQ.3.OR.ICUM.EQ.7) GO TO 350
      PY(NUM)=-(C+A*X(I))/(B+SMAL)
      PX(NUM)=X(I)
      GO TO 340
  350 PX(NUM)=-(C+B*Y(I))/(A+SMAL)
      PY(NUM)=Y(I)
  340 CONTINUE
      IF(NUM.LT.2) GO TO 998
      D1=(PX(1)-XA)**2+(PY(1)-YA)**2
      D2=(PX(2)-XA)**2+(PY(2)-YA)**2
      NUM1=1
      IF(D2.LT.D1) NUM1=2
      XA=PX(NUM1)
      YA=PY(NUM1)
      NUM2=2
      IF(NUM1.EQ.2) NUM2=1
      XB=PX(NUM2)
      YB=PY(NUM2)
      MOD=5
      GO TO 999
  390 CONTINUE
      MOD=-1
  999 CONTINUE
      RETURN
  998 MOD=-2
      GO TO 999
      END
      SUBROUTINE DECOD(PP,VV,AA,JCON,ISYM,IVEC,I)
      COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX
      COMMON/PLBAS2/ AP(16),AV(16),CP(16),DAT(8)
      COMMON/PLBAS3/ WINXL,WINYL,WINXW,WINYW,IWIN
      COMMON/PLBAS4/ SCRNXL,SCRNYL,SCRNXW,SCRNYW,ISCRN
      COMMON/PLBAS5/ SIGNOR,SNPLOT,IH
```

```
      COMMON/PLBAS7/HT,NDECFX,XLATE,YLATE
      DIMENSION PP(3),VV(3)
      IVEC=0
      IF(I.GE.NUMAX) GO TO 999
      IF(I.GT.NUM) GO TO 999
      DO 10 L=1,3
   10 PP(L)=P(L,I)
      AA=P(4,I)
      JCON=ICON(I)/10
      ISYM=ICON(I)-10*JCON
      IF(JCON.GE.5) GO TO 997
      IF(ISYM.GT.3) GO TO 999
      INEX=ICON(I+1)/10
      IVEC=0
      IF(INEX.NE.5) GO TO 998
      I=I+1
      DO 20 L=1,3
   20 VV(L)=P(L,I)
      IVEC=1
  998 CONTINUE
      RETURN
  999 CONTINUE
      I=-1
      RETURN
  997 CONTINUE
      IVEC=999
      IF(JCON.NE.7) RETURN
      IF(ISYM.EQ.1) HT=P(4,I)
      IF(ISYM.EQ.2) NDECFX=P(4,I)
      IF(ISYM.EQ.3) XLATE=P(4,I)
      IF(ISYM.EQ.4) YLATE=P(4,I)
      RETURN
      END
      SUBROUTINE MMULT(A,B,C,M)
C--- CONSTRUCT C=A*B AND STORE THE RESULT IN A OR B
      DIMENSION A(16),B(16),C(16)
      DIMENSION ITEMP(4)
      DATA ITEMP/1,5,9,13/
      DO 10 IROW=1,4
      DO 10 ICOL=1,4
      KK=ITEMP(ICOL)
      SUM=0.0
      DO 11 K=1,4
      SUM=SUM+A(IROW+K*4-4)*B(KK+K-1)
   11 CONTINUE
      C(4*ICOL-4+IROW)=SUM
   10 CONTINUE
      IDEBUG=0
      IF(IDEBUG.EQ.0) GO TO 20
      WRITE(6,50)
   50 FORMAT(//)
      DO 30 I=1,4
      IL=I+12
      WRITE(6,40) (A(L),L=I,IL,4),(B(L),L=I,IL,4),(C(L),L=I,IL,4)
   40 FORMAT(' MMULT',4(1X,F8.3),3X,4(1X,F8.3),3X,4(1X,F8.3))
   30 CONTINUE
   20 CONTINUE
```

```
      IF(M.EQ.3) RETURN
      DO 12 I=1,16
      IF(M.EQ.1) A(I)=C(I)
      IF(M.EQ.2) B(I)=C(I)
   12 CONTINUE
      RETURN
      END
      SUBROUTINE PERSPT(DAT,B)
C--- GENERATE A PROJECTIVE MATRIX B FROM A SIMPLE COMMAND DAT
      DIMENSION DAT(1),B(1),AID(16)
      DATA AID/1.0,4*0.0,1.0,4*0.0,1.0,4*0.0,1.0/
      DATA CDR/0.01745329251994/
C---  DAT(1) CONTAINS THE COMMAND FLAG  =1=XYROT, 2=YZROT,
C---  3=ZXROT, 4=VARIABLE SCALE, 5=TRANS, 6=CENTER
      DO 10 I=1,16
   10 B(I)=AID(I)
      IFLAG=DAT(1)
      IF(IFLAG.GT.3) GO TO 50
      A=DAT(2)*CDR
      C=COS(A)
      S=SIN(A)
      GO TO (20,30,40),IFLAG
   20 B(1)=C
      B(2)=-S
      B(5)=S
      B(6)=C
      GO TO 100
   30 B(6)=C
      B(7)=-S
      B(10)=S
      B(11)=C
      GO TO 100
   40 B(1)=C
      B(3)=S
      B(9)=-S
      B(11)=C
      GO TO 100
   50 IFLAG=IFLAG-3
      GO TO (60,70,80), IFLAG
   60 W=DAT(3)**2+DAT(4)**2
      IF(W.LT.0.000001) GO TO 65
      B(1)=DAT(2)
      B(6)=DAT(3)
      B(11)=DAT(4)
      GO TO 100
   65 B(1)=DAT(2)
      B(6)=DAT(2)
      B(11)=DAT(2)
      GO TO 100
   70 B(4)=DAT(2)
      B(8)=DAT(3)
      B(12)=DAT(4)
      GO TO 100
   80 D=ABS(DAT(2))
      B(11)=0.0
      IF(D.GT.0.0001) B(15)=-1./DAT(2)
      D1=ABS(DAT(3))
```

```
      IF(D1.GT.0.0001.AND.D.GT.0.0001) B(16)= DAT(3)/DAT(2)
      B(4)=-DAT(4)
      B(8)=-DAT(5)
  100 CONTINUE
      IDEBUG=0
      IF(IDEBUG.EQ.0) RETURN
      WRITE(6,140)
  140 FORMAT(//)
      DO 110 I=1,4
      IL=I+12
      WRITE(6,120) (B(J),J=I,IL,4)
  120 FORMAT(10X,'PERSPT',4(2X,F12.5))
  110 CONTINUE
      RETURN
      END
      SUBROUTINE USER
      COMMON/PLBAS1/ P(4,3001),ICON(3001),NUM,NUMAX,IPLTX
      COMMON/PLBAS2/ AP(16),AV(16),CP(16),DAT(8)
      COMMON/PLBAS3/ WINXL,WINYL,WINXW,WINYW,IWIN
      COMMON/PLBAS4/ SCRNXL,SCRNYL,SCRNXW,SCRNYW,ISCRN
      COMMON/PLBAS5/ SIGNOR,SNPLOT,IH
      RETURN
      END
```